# IMPLEMENTASI ENKRIPSI DAN DEKRIPSI ALGORITMA RSA DAN KOMPRESI SHANNON - FANO DALAM PENGAMANAN DATA TEKS

**TUGAS AKHIR**

Diajukan Sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana Teknik Pada
Jurusan Teknik Informatika

**Oleh:**

**JEPRIANTO**

**11351103145**

**FAKULTAS SAINS DAN TEKNOLOGI**
**UNIVERSITAS ISLAM NEGERI SULTAN SYARIF KASIM RIAU**
**PEKANBARU**
**2019**

# LEMBAR PERSETUJUAN

# IMPLEMENTASI ENKRIPSI DAN DEKRIPSI ALGORITMA RSA DAN KOMPRESI SHANNON - FANO DALAM PENGAMANAN DATA TEKS

## TUGAS AKHIR

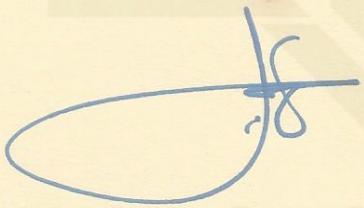
**Oleh:**

**JEPRIANTO**

**11351103145**

Telah diperiksa dan disetujui sebagai laporan tugas akhir
di Pekanbaru pada tanggal Oktober 2019

Pembimbing,



**Febi Yanto, M.Kom**

**NIP.198102062009121003**

# LEMBAR PENGESAHAN

## IMPLEMENTASI ENKRIPSI DAN DEKRIPSI ALGORITMA RSA DAN KOMPRESI SHANNON - FANO DALAM PENGAMANAN DATA TEKS

## TUGAS AKHIR

Oleh

**JEPRIANTO**
**11351103145**

Telah dipertahankan di depan sidang dewan penguji
Sebagai salah satu syarat untuk memperoleh gelar sarjana Teknik Informatika
Fakultas Sains dan Teknologi Universitas Islam Negeri Sultan Syarif Kasim Riau
di Pekanbaru, pada tanggal Oktober 2019

Pekanbaru, Oktober 2019

Mengesahkan,

| Dekan | Ketua Jurusan, |
|---|---|
| **Dr. Drs. H. Mas'ud Zein, M.Pd** | **Dr. Elin Haerani, S.T, M.Kom** |
| **NIP. 19631214 198803 1 002** | **NIP. 19810523 200710 2 003** |

DEWAN PENGUJI

| | |
|---|---|
| Ketua | : Dr. Elin Haerani, S.T, M.Kom |
| Sekretaris | : Febi Yanto, M.Kom |
| Penguji I | : Reski Mai Candra, S.T, M.Sc |
| Penguji II | : Pizaini, S.T, M.Kom |

# LEMBAR HAK ATAS KEKAYAAN INTELEKTUAL

Tugas Akhir yang tidak diterbitkan ini terdaftar dan tersedia di Perpustakaan Universitas Islam Negeri Sultan Syarif Kasim Riau adalah terbuka untuk umum dengan ketentuan bahwa hak cipta pada penulis. Referensi kepustakaan diperkenankan dicatat, tetapi pengutipan atau ringkasan hanya dapat dilakukan seizin penulis dan harus disertai dengan kebiasaan ilmiah untuk menyebutkan sumbernya.

Penggandaan atau penerbitan sebagian atau seluruh Tugas Akhir ini harus memperoleh izin dari Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Sultan Syarif Kasim Riau. Perpustakaan yang meminjamkan Tugas Akhir ini untuk anggotanya diharapkan untuk mengisi nama, tanda peminjaman dan tanggal pinjam.

# LEMBAR PERNYATAAN

Dengan ini saya menyatakan bahwa dalam Tugas Akhir ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan didalam daftar pustaka.

Pekanbaru, Oktober 2019

Yang membuat pernyataan,

**JEPRIANTO**

# LEMBAR PERSEMBAHAN

بِسْمِ اللَّهِ الرَّحْمَٰنِ الرَّحِيمِ

فَإِنَّ مَعَ الْعُسْرِ يُسْرًا ۝٥

*" Karena sesungguhnya sesudah kesulitan itu ada kemudahan ".*

**(QS.Alam Nasyrah(5))**

***

**Hasil Tugas Akhir ini penulis persembahkan untuk**

**Ayah dan ibu penulis yang selalu memberikan do'a dan dukungan yang tak pernah putus.**

**Kemudian, untuk keluarga besar penulis.**

**Untuk teman-teman penulis**

**Serta terima kasih untuk keluarga besar TIFE 13 dan semua orang yang selalu memberi semangat.**

***

# IMPLEMENTASI ENKRIPSI DAN DEKRIPSI ALGORITMA RSA DAN KOMPRESI SHANNON - FANO DALAM PENGAMANAN DATA TEKS

**JEPRIANTO**

**11351103145**

Jurusan Teknik Informatika

Fakultas Sains Dan Teknologi

Universitas Sultan Syarif Kasim Riau

## ABSTRAK

Keamanan dan kerahasiaan data merupakan aspek yang sangat penting untuk dijaga. Pada kasus yang telah terjadi dengan cara memasukkan aplikasi kedalam perangkat pengunjung yang dapat mengekstrak email, kontak dan file pengunjung yang datang. Untuk melakukan pengamanan data teks dengan melakukan enkripsi dan kompresi menggunakan algoritma RSA dan Shannon-Fano. Data yang digunakan adalah teks atau file teks. Teks atau data teks akan di enkripsi terlebih dahulu dengan algoritma RSA, setelah dienkripsi akan dilakukan kompresi dengan algoritma Shannon-Fano. Dengan rata rata kecepatan pembangkitan kunci 0,155 detik, kecepatan enkripsi 0,020 detik, kecepatan kompresi 0,017 detik, kecepatan dekompresi 0,044 detik dan kecepatan dekripsi 0,039 detik dan hasil analisa perhitungan kompresi data teks, teks sebelum di kompresi dengan ukuran 96 bit menjadi 52 bit setelah di kompresi. Dengan melakukan enkripsi dan kompresi data teks atau file teks maka telah membantu dalam berkirim pesan rahasia yang tidak ingin dilihat oleh pihak ketiga.

**Kata Kunci : Enkripsi, Kompresi, RSA, Shannon-Fano, Teks**

# *IMPLEMENTATION OF RSA ALGORITHM AND SHANNON-FANO COMPRESSION ENCRYPTION AND COMPRESSION IN TEXT DATA SECURITY*

**JEPRIANTO**

**11351103145**

*Informatics Engineering*

*Faculty of Science and Technology*

*Sultan Syarif Kasim Riau University*

## *ABSTRACT*

*Data security and confidentiality are very important aspects to protect. In cases that have occurred by entering the application into the visitor's device that can extract emails, contacts and files of visitors who come. To perform text data security by doing encryption and compression using RSA and Shannon-Fano algorithms. The data used is text or text file. Text or text data will be encrypted first with the RSA algorithm, after encryption the compression will be performed with the Shannon-Fano algorithm. With an average key generation speed of 0.155 seconds, an encryption speed of 0.020 seconds, a compression speed of 0.017 seconds, a decompression speed of 0.044 seconds and a decryption speed of 0.039 seconds and the results of the analysis of text data compression calculations, the text before being compressed with the size of 96 bits to 52 bits after compression . Encrypting and compressing text data or text files helps to send secret messages that third parties do not want to see.*

***Keywords: Encryption, Compression, RSA, Shannon-Fano, Text***

# KATA PENGANTAR

*Assalammu'alaikum Warahmatullahi Wabarakatuh.*

*Alhamdulillaahi Robbil'alamin*, puji dan syukur yang setinggi-tinggi penulis ucapkan ke-hadirat Allah SWT, karena atas segala limpahan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan penelitian sekaligus penulisan laporan tugas akhir ini. *Allahumma sholli'ala Muhammad wa'ala ali sayyidina Muhammad*, tidak lupa penulis haturkan juga untuk junjungan alam, kekasih Allah, Rasul Allah, dan suri tauladan kita yakni Nabi Muhammad SAW.

Selama menyelesaikan tugas akhir ini, penulis telah banyak mendapatkan bantuan, bimbingan, dan petunjuk dari banyak pihak baik secara langsung maupun tidak langsung. Dalam kesempatan ini penulis ingin mengucapkan terimakasih yang sebesar-besarnya kepada:

1. Bapak Prof. Dr. H. Akhmad Mujahidin, S.Ag, M.Ag, selaku Rektor Universitas Islam Negeri Sultan Syarif Kasim Riau.
2. Bapak Dr. Drs. H. Mas'ud Zein, selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Sultan Syarif Kasim Riau.
3. Ibu Dr. Elin Haerani, S.T, M.Kom selaku Ketua Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Sultan Syarif Kasim Riau.
4. Bapak Novriyanto, S.T, M.Sc, selaku Pembimbing Akademis penulis.
5. Ibu Sonya Meitarice, S.T, M.Sc, selaku Koordinator Tugas Akhir.
6. Bapak Febi Yanto, M.Kom, selaku Pembimbing Tugas Akhir penulis. Terimakasih banyak kepada beliau atas waktu, kesempatan, arahan, dorongan, saran, ilmu dan petunjuk yang telah diberikan. Atas semua itu, penulis dapat memulai hingga menyelesaikan tugas akhir ini dengan baik.
7. Bapak Reski Mai Candra, S.T,M.Sc, selaku penguji I, terimakasih pak untuk ilmu-ilmunya, saran-sarannya, perbaikan-perbaikannya, dan masukan yang bapak berikan untuk penyempurnaan tugas akhir ini.

8. Bapak Pizaini, S.T, M.Kom, selaku penguji II, terimakasih pak untuk ilmu-ilmunya, saran-sarannya, perbaikan-perbaikannya, dan masukan yang bapak berikan untuk penyempurnaan tugas akhir ini.
9. Seluruh Dosen Teknik Informatika yang telah memberikan ilmunya kepada penulis, hingga akhirnya penulis dapat menyelesaikan kuliah di jenjang S1 ini. Terimakasih untuk bapak dan ibu, penulis tidak pernah dapat melupakan jasa beliau semua.
10. Ayah dan Ibu penulis serta adik adik. Terimakasih telah selalu menjadi pahlawan, guru serta penyemangat terhebat untuk kesuksesan dunia dan akhirat. Terimakasih banyak untuk doa kalian yang tidak pernah berhenti untuk selalu menjadikan kami anak yang sukses.
11. Teman seperjuangan yang selama pengerjaan tugas akhir ini selalu bersama dalam suka maupun duka Thomas Perguso Syurahman, Yusuf Abdillah Putra, Syarif Hidayatullah, Jasriadi, Rivalza Fahlevi, Eko Waluyo Panji Sukoco, Fakhrial dan teman teman lainnya khususnya TIF E 13, walaupun kalian membuat saya malas kekampus dan mematahkan semangat saya pergi kekampus untuk bimbingan. Akan tetapi kalian tetap menemani dalam pengerjaan tugas akhir ini.
12. Semua pihak yang terlibat baik langsung maupun tidak langsung dalam pelaksanaan tugas akhir dan tidak dapat penulis sebutkan satu persatu.

Penulis menyadari bahwa dalam penulisan laporan tugas akhir ini masih banyak kesalahan dan kekurangan. Kritik dan saran yang membangun sangat penulis harapkan untuk kesempurnaan laporan ini. Semoga laporan ini dapat memberikan manfaat bagi pembaca. Âamiin yaa Rabbal'âlaamîin.

*Wassalâmu'alaikum wa rahmatullâhi wa barakâtuh*

Pekanbaru, Oktober 2019

Jeprianto

# DAFTAR ISI

# DAFTAR GAMBAR

Halaman

# DAFTAR TABEL

Halaman

# DAFTAR RUMUS

Halaman

# DAFTAR DIAGRAM

Halaman

# DAFTAR SIMBOL

*: Start Event,* adalah symbol yang mengidentifikasi sebuah proses yang akan dimulai.

: *Activity,* adalah sebuah istilah umum untuk mewakili semua kegiatan sehari-hari.

*: Gateway,* dapat didefinisikan sebagai semua tindakan arus urutan proses bisnis. *Gateway* kadang memainkan salah satu dari dua peran.

: *End Event,* diindikasikan sebagai simbol untuk mengakhiri sebuah proses.

: *Connection Object*, adalah elemen yang menghubungkan flow objek.

# BAB I
# PENDAHULUAN

## 1.1 Latar Belakang

Keamanan dan kerahasiaan data merupakan aspek yang sangat penting untuk dijaga. Pentingnya informasi yang dikirim dan diterima oleh orang bersangkutan dikarenakan informasi membutuhkan privasi atau kerahasiaan dan informasi tidak akan berguna lagi jika informasi tersebut telah diketahui oleh pihak ketiga (Infotama, 2010).

Kriptografi merupakan cara mengamankan pesan dan data yang ingin disembunyikan dari pihak yang tidak terkait. Dengan melakukan enkripsi pesan maka pesan tersebut akan disembunyikan dengan metode yang ada dalam kriptografi. Dekripsi pesan lah yang akan membuka kembali pesan yang telah disembunyikan tadi dengan kunci yang telah disediakan (Zulham, 2014).

Banyak data teks ataupun gambar yang harus diamankan dari pihak ketiga dikarenakan data itu merupakan data privasi atau rahasia yang sangat penting. Data teks atau gambar sangat mudah diketahui apabila dalam bentuk fisik . Data teks atau gambar dapat diamankan dengan melakukan enkripsi dan dapat membukanya kembali dengan melakukan dekripsi.

Pada kasus yang di liput oleh (Suriawati, 2019) pada lama Rakyatku News, dengan hasil bahwa penjaga perbatasan cina memasukkan aplikasi khusus untuk melakukan penyadapan atau pengawasan di ponsel pengunjung yang memasuki Kawasan tersebut. Secara khusus aplikasi tersebut mengekstrak email, pesan teks serta informasi kontak yang ada pada ponsel tersebut.

Ada banyak sekali cara dalam melakukan pengamanan data seperti gambar, suara maupun tulisan atau teks. Ada banyak metode dan algoritma untuk melakukan enkripsi dan dekripsinya. Seperti algoritma RSA, RSA merupakan algoritma asimetris atau enkripsi dengan menggunakan dua kunci (*public* dan *private*). Sudah banyak penelitian yang meneliti menggunakan algoritma RSA dalam pengamanan

data teks ataupun citra atau gambar. Maka dari itu pada penelitian ini dibuat untuk melakukan pengamanan data teks dengan menggunakan algoritma RSA dan kompresi Shannon - Fano .

Dalam beberapa penelitian yang meneliti penggunaan algoritma RSA seperti pada penelitian yang dilakukan (Ginting, Isnanto dan Windasari, 2015) dengan penelitian implementasi algoritma kriptografi RSA pada *e-mail* pada penelitian ini menjelaskan tentang penggunaan algoritma RSA untuk melakukan enkripsi dan pengamanan pesan pada *e-mail*. Pada penelitian ini algoritma yang dipakai sudah baik dalam melakukan enkripsi akan tetapi kunci yang dipakai terlalu kecil hingga membuat keamanannya berkurang.

Dalam penelitian (Wulansari, Setyawan, & Susanto, 2016) berikutnya ada juga yang meneliti mengukur kecepatan enkripsi dan dekripsi algoritma RSA pada pengembangan sistem informasi *text security* , pada penelitian ini membandingkan kecepatan logika algoritma dalam melakukan enkripsi data teks. Pada penelitian ini hanya melihat kecepatan enkripsi akan tetapi tidak melihat kualitas data yang diperoleh dari enkripsi tersebut.

Dalam penelitian (Asmoro, 2015) juga meneliti pengamanan file citra atau gambar, meneliti dengan menggunakan algoritma RSA dan otp dalam penggabungannya. Dalam penelitian ini gambar di enkripsi dengan RSA lalu di enkripsi kembali dengan otp. Pada penelitian ini keamanan file sudah kuat, akan tetapi waktu yang dipakai cukup lama.

Dan pada penelitian yang dilakukan oleh (Christine Lamorahan, Benny Pinontoan, 2013) meneliti tentang perbandingan kompresi Shannon – Fano dan kompresi Huffman, mendapatkan hasil kompresi Shannon – Fano lebih unggul dibandingkan kompresi Huffman ketika data yang berukuran besar (Christine Lamorahan, Benny Pinontoan, 2013).

Berdasarkan dari itu dibuatlah penelitian tentang enkripsi dan dekripsi data teks dengan menggunakan algoritma RSA dan kompresi Shannon - Fano. Ingin menguji keamanan dan juga melihat ukuran data teks yang sudah di enkripsi dan juga dikompress dengan menggunakan algoritma RSA dan kompresi Shannon - Fano.

## 1.2 Rumusan Masalah

Rumusan masalah pada penelitian ini adalah bagaimana merancang dan membangun sistem atau aplikasi pengamanan data teks dengan menggunakan algoritma RSA dan kompresi Shannon – Fano.

## 1.3 Batasan Masalah

Adapun batasan masalah pada penelitian ini adalah sebagai berikut:,

1. Menggunakan huruf, karakter, symbol sebagai data uji yang akan diinputkan.
2. Memberikan batasan rentang nilai P dan Q dalam pembangkitan kunci. Dengan rentang nilai 0 – 1.000.000

## 1.4 Tujuan Penelitian

Tujuan penelitian ini adalah untuk merancang dan membangun sistem atau aplikasi pengamanan data teks dengan menggunakan algoritma RSA dalam melakukan enkripsi dan enkripsi dan kompress Shannon - Fano dalam mengompress teks setelah di enkripsi.

## 1.5 Sistematika Penulisan

Berikut merupakan rencana susunan sistematika penulisan laporan tugas akhir yang akan dibuat:

**BAB I PENDAHULUAN**

Bab ini menjelaskan tentang deskripsi umum tugas akhir yang meliputi latar belakang, rumusan masalah, batasan masalah, tujuan penelitian dan sistematika penulisan.

**BAB II LANDASAN TEORI**

Bab ini menjelaskan tentang tentang kriptografi, terminology kriptografi, jenis-jenis kriptografi, jenis serangan kriptografi, tujuan kriptografi, teks, algoritma RSA, kompresi Shannon - Fano, dan penelitian sebelumnya.

**BAB III METODOLOGI PENELITIAN**

Bab ini menjelaskan tentang metodologi yang dipakai dalam penelitian, identifikasi masalah, pengumpulan data, analisa dan perancangan, implementasi dan pengujian, kesimpulan dan saran.

**BAB IV ANALISA DAN PERANCANGAN**

Bab ini menjelaskan tentang analisa dan perancangan dimulai dari analisa komponen system, analisa tampilan system, serta bagaimana algoritma RSA bekerja dalam sistem.

**BAB V IMPLEMENTASI DAN PENGUJIAN**

Bab ini menjelaskan tentang implementasi dan pengujian aplikasi yang dibangun mulai dari pengujian elemen aplikasi, pengujian kecepatan, pengujian ukuran data, pengujian *recovery*.

**BAB VI PENUTUP**

Bab ini menjelaskan tentang kesimpulan dan saran.

# BAB II
# LANDASAN TEORI

## 2.1 Kriptografi

Kata kriptografi pertama kali dikenalkan pada zaman romawi kuno oleh Julius Caesar untuk keperluan komunikasi pada satuan militernya. Akan tetepai sekutu dpaat menembusnya. Sekutu akhirnya memenangkan perang dunia kedua dikarenakan mengetahui startegi mereka.

Kriptografi berasal dari Bahasa yunani cryptos artinya *secret* atau rahasia, sedangkan graphein artinya *writing* atau tulisan. Jadi kriptografi berarti *secret writing* atau tulisan rahasia. Ada banyak penjabaran kriptografi yang telah diutarakan sekitar pada tahun 1980 dan mengatakan kriptografi adalah ilmu atau cara menyembunyikan pesan yang ingin dikunci agar tidak dapat terlihat oleh orang lain yang tidak diinginkan. Tetapi seiring perkembangan zaman hingga saat ini pengertian kriptografi berkembang menjadi ilmu tentang teknik matematika yang digunakan untuk menyelesaikan persoalan keamanan berupa privasi dan otentikasi (Diffie, 1976) dan juga mendefinisikan bahawa kriptografi adalah ilmu yang memepelajari teknik matematika yang berhubungan dengan aspek keamanan informasi seperti kerahasiaan, integritas pesan dan otentikasinya.

### 2.1.1 Terminologi Kriptografi

Adapun terminologi pada kriptografi adalah sebagai berikut :

a. Pesan, plainteks dan cipherteks

   Pesan adalah informasi yang dapat dibaca oleh seseorang. Plainteks merupakan pesan asli yang akan dilakukan enkripsi , sedangkan cipherteks merupakan pesan hasil enkripsi yang berasal dari plainteks atau pesan awal (Munir, 2006).

b. Pengirim dan penerima

   Komunikasi yang melibatkan antara pengirim dan penerima. Pengirim merupakan orang yang akan mengirim pesan yang sudah dijadikan cipherteks

sedangkan penerima merupakan orang yang akan menerima cipherteks dari pengirim untuk dijadikan plainteks (Munir, 2006).

c. Enkripsi dan dekripsi

Enkripsi merupakan langkah untuk menjadika plainteks menjadi cipherteks sedangkan dekripsi merupakan langkah untuk menjadikan cipherteks menjadi plainteks awal. Enkripsi dan dekripsi dapat diterapkan pada pesan yang dikirim dan tersimpan. Istilah *encryption of data in motion* mengacu pada enkripsi pesan yang ditransmisikan melalui saluran komunikasi, sedangkan istilah *encryption of data at-rest* mengacu pada enkripsi dokumen yang disimpan didalam storage (Munir, 2006). Proses enkripsi dan dekripsi dapat dilihat pada gambar 2.1 :


**Gambar 2.1 Enkripsi dan Dekripsi**

(Sumber : http://widuri.raharja.info)

d. Cipher dan kunci

Cipher merupakan cara atau fungsi matematika yang digunakan untuk melakukan enkripsi dan dekripsi. Beberapa cipher memerlukan algoritma yang berbeda untuk melakukan enkripsi dan dekripsi.
(Munir, 2006).

e. Sistem kriptografi

Kriptografi membentuk sebuah sistem yang menjadi system kriptografi, didalam system kriptografi terdiri dari algoritma dalam melakukan enkrips dan denkripsi (Munir, 2006).

f. Penyadap

Penyadap adalah orang yang mencoba untuk mencuri pesan yang dikirim kemudian mencoba membuka pesan tersebut denga berbagai cara (Munir, 2006).

g. Kriptanalis dan kriptologi

Kriptanalis adalah ilmu ataupun seni untuk melakukan pembukaan cipherteks tanpa harus mengetahui kuncinya, biasanya akan dilakukan dengan cara apapun (Munir, 2006).

### 2.1.2 Jenis – Jenis Kriptografi

Algoritma kriptografi dapat diklasifikasikan menjadi menjadi dua jenis berdasarkan perkembangannya, yaitu kriptografi klasik dan kriptografi modern.

a. Algoritma Kriptgrafi Klasik

Algoritma inisudah ada sejak lama sekali biasanya dengan teknik substitusi atau transposisi atau keduanya (Sadikin, 2012). Teknik substitusi adalah menggantikan karakter plainteks dengan karakter lain sehingga menghasil cipherteks. Sedangkan transposisi adalah cara mengubah *plaintext* menjadi *ciphertext* dengan cara permutasi. (Prayudi, 2005).

b. Algoritma Kriptografi Modern

Algoritma ini memiliki tingkat kesulitan yang cukup rumit (Prayudi, 2005), dan kekuatan kriptografinya terletak pada kuncinya (Wirdasari, 2008). Algoritma ini menggunakan simbol biner karena berjalan mengikuti operasi komputer digital. Sehingga membutuhkan dasar berupa pengetahuan matematika untuk mengetahuinya (Sadikin, 2012).

Jenis kriptografi berdasarkan kunci yang dipakai dapat dibedakan sebagai berikut :

a. Algoritma Simetris

Algoritma ini disebut simetris karena memiliki kunci yang sama untuk proses enkripsi dan dekripsinya (Prayudi, 2005).

b. Algoritma Asimetris

Algoritma ini disebut asimetris karena kunci yang digunakan ada dua yaitu kunci privat dan kunci public (Prayudi, 2005).

### 2.1.3 Tujuan Kriptografi

Ada empat tujuan mendasar dari ilmu kriptografi ini yang juga merupakan aspek keamanan informasi (Munir, 2006) yaitu :

1. Kerahasiaan
2. Integritas data
3. Autentikasi
4. Non Repudiasi

### 2.1.4 Jenis Serangan Kriptografi

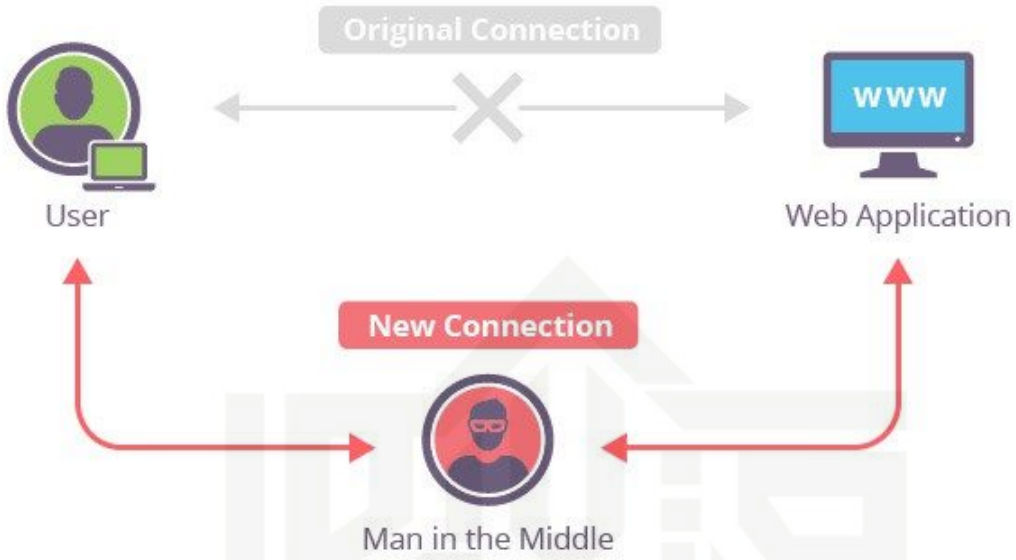
Berdasarkan keterlibatan penyerang dalam melakukan kegiatannya (Munir, 2006) :

1. Serangan pasif

   Penyerang tidak terlibat dalam komunikasi antara pengirim dan penerima, penyerang hanya melakukan penyadapan untuk memperoleh data. Metode yang digunakan dalam melakukan penyadapan ini biasanya *wiretapping, electromagnetic eavesdropping* atau *acoustic eavesdropping*.

2. Serangan aktif

   Penyerang mengintervensi komunikasi dan ikut mempengaruhi untuk keuntungannya. Penyerang mengubah aliran pesan seperti menghapus sebagian *ciphertext*, mengubah *ciphertext*, menyisipkan potongan *ciphertext* palsu, mengubah informasi yang tersimpan, dan lain sebagainya. Metode yang digunakan adalah *Man-in-the-middle-attack* **,** penyerang mengaku seolah-olah sebagai pihak yang berhak menerima pesan atas pesan yang dikirim. Atau sebaliknya penyerang bertindak seolah-olah sebagai pemberi pesan yang asli. Dapat dilihat pada gambar 2.2:




**Gambar 2.2 *Man Middle Attack***

(Sumber : https://www.imperva.com)

Pengetahuan mengenai serangan terhadap kriptografi sangatlah penting untuk meningkatkan efektifitas dan kualitas algoritma yang digunakan. Prinsip yang dipakai dalam menentukan penggunaan suatu algoritma kriptografi adalah :

1. Persamaan matematis yang menggambarkan operasi algoritma kriptografi yang dibuat sangat terstruktur sehingga algoritma tidak mungkin dipecahkan secara analitik.
2. Biaya untuk memecahkan *ciphertext* melampaui nilai informasi yang terdapat di dalam *ciphertext* tersebut.
3. Waktu yang diperlukan untuk memecahkan *ciphertext* tersebut melampaui lamanya waktu informasi tersebut dalam penjagaan kerahasiaannya.

Ada beberapa metode penyerangan yang dapat dilakukan *attacker*, sebagai berikut :

1. Hanya *ciphertext* yang diketahui

   Kriptanalis hanya memiliki *ciphertext* tanpa memiliki *plaintext* nya. Sebelum melakukan serangan, kriptanalis selalu membuat dugaan algoritma yang digunakan dalam pembentukan *ciphertext* itu untuk menentukan cara memecahkannya.

   Teknik yang digunakan untuk menemukan *plaintext*/kunci :

a. Analisa frekuensi huruf

Setiap bahasa memiliki ciri atas huruf. Ciri ini merupakan frekuensi dalam penulisan huruf tersebut. Teknik ini umumnya digunakan untuk memecahkan metode penyandian sederhana seperti seperti model Caesar.

*b. Exhausted Key Search*

Yaitu cara mengetehui plainteks dengan cara mencoba kemungkinan kunci yang ada dan meneba nebak kemungkinannya. Tabel berikut memperlihatkan waktu yang dibutuhkan untuk exhaustive key search :

**Tabel : 2.1 *Exhaustive Key Search***

| Ukuran Kunci | Jumlah Kemungkinan Kunci | Lama Wakti Untuk $10^6$ Percobaan Per Detik | Lama Wakti Untuk $10^{12}$ Percobaan Per Detik |
|---|---|---|---|
| 16 bit | $2^{16} = 65536$ | 32.7 milidetik | 0.0327 mikrodetik |
| 32 bit | $2^{32} = 4.3^{x}109$ | 35.8 menit | 2.15 milidetik |
| 56 bit | $2^{56} = 7.2^{x}1016$ | 1142 tahun | 10.01 jam |
| 128 bit | $2^{128} = 4.3^{x}1038$ | $5.4^{x}1024$ tahun | $5.4^{x}1018$ tahun |

(Sumber : William Stallings, *Data and Computer Communication Fourth Edition)*

c. *Analytical attack*

Yaitu teknik memecahkan teks sandi dengan melakukan anlisa kelemahan dari algoritma tesebut, dengan penggabungan *Exhasuted Key Search* maka akan mempercepat kemungkinannya.

2. *Ciphertext* terpilih

Kriptanalis memilih *ciphertext*, dan kemudian melalui *ciphertext* itu berusaha untuk mendapatkan *plaintext* yang sesuai. Biasanya dilakukan untuk menyerang kriptografi sistem kunci publik.

3. *Plaintext* dan *ciphertext* diketahui

Kriptanalis mempunyai baik *plaintext* maupun *ciphertext*-nya dan melakukan pencarian kesamaan untuk mengetahui kunci pembentuknya. Beberapa pesan biasanya terdapat format yang sudah terstruktur. Format ini merupakan celah yang membuka peluang untuk menerka *ciphertext* dari *plaintext*.

4. Kunci terpilih

Kriptanalis memiliki pengetahuan mengenai hubungan antara kunci-kunci yang berbeda, dan kemudian memilih kunci yang tepat untuk membuka pesan. Biasanya digunakan untuk mengetahui algoritma penyandian suatu pesan.

5. *Social engineering*

Mencari informasi algoritma atau kunci dengan cara melaukan penipuan, pemerasan terhadap sipemegang kunci sampai orang yang memegang kunci memberinya kunci untuk membuka pesan.

## 2.2 Kompresi

Kompresi adalah proses mengkonversikan sebuah data inputan yaitu data stream atau data mentah menjadi data stream lainnya atau data asli yang sudah terkompresi yang berukuran lebih kecil dari sebelumnya (Salomon, 2007). kompresi data membuat sebuah aliran data inputan kedalam aliran data yang lain yang memiliki ukuran yang lebih kecil.

Tujuan kompresi adalah mempresentasikan data digital dengan sedikit bit dan tetap mempertahankan keutuhan atau kebutuhan kamsimum untuk membentuk kembali data aslinya. data digital ini dapat berupa teks, gambar, audio maupun video. contoh kompresi yang sering kita gunakan dalam kehidupoan sehari hari adalah membuat kata “bukan” dengan mengecilkannya menjadi “bkn”.

### 2.2.1 Jenis – Jenis Kompresi

Secara garis besar, terdapat 2 penggolongan algoritma kompresi data: kompresi lossy dan kompresi lossless.

1. Kompresi Lossy

Algoritma kompresi dikatakan lossy jika tidak dimungkinkan untuk membentuk data asli yang tepat sama dari data yang sudah dikompresi. Ada beberapa data yang hilang selama proses kompresi. Contoh penggunaan algoritma lossy seperti pada data gambar, suara dan video. Karena cara kerja sistem pengelihatan dan pendengaran manusia yang minim, beberapa data dapat

dihilangkan, sehingga didapat data hasil kompresi yang seolah-olah sama dengan data yang asli.

2. Kompresi Lossless

Algoritma kompresi dikatakan lossless jika dimungkinkan untuk membentuk data asli yang tepat sama dari data yang sudah dikompresi. Tidak ada informasi yang hilang selama proses kompresi dan dekompresi. Teknik ini digunakan jika data tersebut sangat penting, jadi tidak di mungkinkan untuk menghilangkan beberapa detail. Untuk kompresi Lossless, berdasarkan cara mereduksi data yang akan dikompresi, terbagi lagi menjadi 2 kelompok besar algoritma:

a. Algoritma Berbasis Entropi

Algoritma berbasis Entropi, atau disebut juga berbasis statistik, menggunakan model statistik dan probabilitas untuk memodelkan data, keefisienan kompresi bergantung kepada berapa banyak karakter yang digunakan dan seberapa besar distribusi probabilitas pada data asli. Contoh algoritma yang berbasis entropi adalah: Huffman Coding, Adaptive Huffman, dan Shannon Fano, Run Length.

b. Algoritma Berbasis Kamus

Algoritma berbasis kamus, bekerja dengan cara menyimpan pola masukan sebelumnya, dan menggunakan index dalam mengakses pola tersebut jika terdapat perulangan. Contoh algoritma yang berbasis dictionary adalah: LZ77, LZ78, LZW, DEFLATE, dan LZMA.

### 2.2.2 Rasio Kompresi

Efek pemampatan pada kompresi lossless dapat diukur melalui jumlah penyusutan suatu file asal dengan membandingkan ukuran dari penyusutan dan pemampatan. Jika dimisalkan rasio pemampatan adalah r, ukuran file setelah pemampatan atau kompresi adalah p dan ukuran file sebelum dikompresi adalah q maka rasio pemampatan dapat dirumuskan sebagai berikut :

$$Rasio\ Kompresi : \frac{Ukuran\ File\ Setelah\ Kompresi}{Ukuran\ File\ Sebelum\ diKompresi} \text{x} 100\% \quad \ldots\ldots\ldots\ldots (2.1)$$

Semakin kecil rasio kompresi maka semaikin tinggi juga pemampatan pada file tersebut (Sutardi, 2014).

## 2.3 Teks

Telah disebutkan dalam buku (Maryanto, Anik Muslikah Indriastuti, 2014), Teks adalah satuan lingual yang dimediakan secara tulis ataupun lisan dengan tata organisasi tertentu untuk mengungkapkan makna secara konteks. Istilah teks dan wacana dianggap sama dan hanya dibedakan dalam hal bahwa wacana lebih bersifat abstrak dan merupakan realisasi makna dari teks. Jenis-jenis teks yang secara umum dikenal adalah deskripsi, laporan, prosedur, penceritaan, eksplanasi, eksposisi, diskusi, surat, editorial, iklan, negosiasi, anekdot, naratif, eksemplum, dan lain-lain. Jenis-jenis teks tersebut mempunyai struktur teks yang berbeda dan memanfaatkan bentuk-bentuk bahasa yang berbeda (misalnya, jenis verba, konjungsi, partisipan, dan kelompok kata). Struktur teks dan bentuk-bentuk bahasa itu menjadi ciri-ciri yang menandai teks-teks tersebut.

### 2.3.1 Struktur Teks

Struktur teks adalah tata organisasi teks, yaitu cara teks disusun. Sebuah teks ditata sesuai dengan jenisnya. Misalnya, teks prosedur mempunyai struktur teks tujuan yang akan dicapai, teks laporan mempunyai struktur teks pernyataan umum atau klasifikasi yang dilaporkan (Maryanto, Anik Muslikah Indriastuti, 2014).

### 2.3.2 Jenis – Jenis Teks

Adapun beberapa jeni teks yang ada, meliputi :

a. Teks Anekdot

   Adalah teks yang berisi peristiwa menggemaskan yang ditanggapi dengan sebuah reaksi.

b. Teks Dekripsi

   Adalah teks yang menggambarkan keadaan suatu barang, karakter atau manusia. Teks ini mengutamakan hubungan antar bagiannya.

c. Teks Diskusi

   Adalah teks yang berisi tinjauan terhadap sebuah permasalahan. Berada diantara pro dan kontra untuk mencapai kesepakatan bersama.

d. Teks Editorial

Adalah teks pada koran atau majalah yang merupakan gagasan terhadap suatu hal yang berkaitan dengan koran tersebut..

e. Teks Eksemplum

Adalah teks rekaan yang berisi insiden yang menurutnya tidak perlu terjadi. Secara pribadi, seseorang menginginkan insiden itu dapat diatasi, tetapi ia tidak dapat berbuat banyak.

f. Teks Eksplanasi

Adalah teks yang menjelaskan hubungan logis dari beberapa peristiwa. Pada teks eksplanasi, sebuah peristiwa timbul karena ada peristiwa lain sebelumnya dan peristiwa tersebut mengakibatkan peristiwa yang lain lagi sesudahnya.

g. Teks Eksposisi

Jenis teks yang berfungsi untuk mengungkapkan gagasan dengan argmentasi yang kuat. Teks ini hanya memandang dari satu sisi saja.

h. Teks Naratif

Teks rekaan yang berisi komplikasi yang menimbulkan masalah yang memerlukan waktu untuk melakukan evaluasi agar dapat memecahkan masalah tersebut. Teks naratif umumnya dijumpai pada dongeng, hikayat, cerita pendek, atau novel.

i. Teks Negosiasi

Adalah proses tawar-menawar dengan jalan berunding guna mencapai kesepakatan bersama antara pihak dan pihak yang lain.

j. Penceritaan

Adalah teks yang berisi pengungkapan pengalaman atau peristiwa yang dilakukan pada masa lalu seseorang.

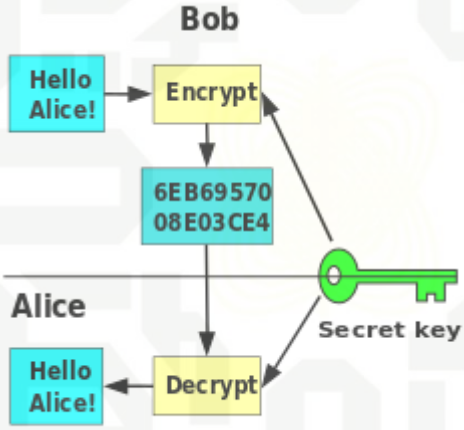
k. Teks Prosedural

Adalah teks yang berisi langkah-langkah yang harus ditempuh untuk mencapai tujuan yang diinginkan. Langkah-langkah itu biasanya tidak dapat dibalik-balik.

## 2.4 Kriptografi RSA

Algoritma RSA merupakan salah satu algoritma asimetris yang populer dipakai dan bahkan masih dipakai hingga saat ini. Kekuatan algoritma ini terletak pada proses eksponensial, dan pemfaktoran bilangan menjadi 2 bilangan prima yang hingga kini perlu waktu yang lama untuk melakukan pemfaktorannya (Wibowo, Susanto, & Shamir, 2009).

Algoritma ini dinamakan sesuai dengan nama penemunya, Ron Rivest, Adi Shamir dan Adleman (Rivest-Shamir-Adleman) yang dipublikasikan pada tahun 1977 di MIT, menjawab tantangan yang diberikan algoritma pertukaran kunci Diffie Hellman, berikut skema RSA :




**Gambar 2.3 Skema RSA**

(sumber : https://id.wikipedia.org/wiki/Kriptografi )

Skema RSA ini menyandang dari skema block cipher, dimana sebelum dilakukannya enkripsi, plainteks yang ada akan dibagi – bagi menjadi beberapa blok dengan panjang yang sama, dimana plainteks dan cipherteksnya berupa bilangan bulat antara 1 hingga n, dimana n berukuran biasanya sebesar 1024 bit, dan panjang bloknya sendiri biasanya berukuran lebih kecil atau sama dengan log(n) +1 dengan basis 2. Fungsi enkripsi dan dekripsinya dijabarkan dalam fungsi berikut :

$$C = Me \bmod n \ (Fungsi\ Enkripsi) \quad \text{.....(2.2)}$$

$$M = Cd \bmod n \ (Fungsi\ Dekripsi) \quad \text{.....(2.3)}$$

Penjelasan :

C = Cipherteks

M = Message / Plainteks

e = Kunci Publik

d= Kunci Privat

n = Modulo pembagi

Kedua pihak harus mengetahui nilai e dan nilai n, dan salah satu pihak harus memilki nilai d untuk melakukan dekripsi terhadap hasil enkripsi dengan menggunakan kunci publik e. Penggunaan algoritma ini harus memenuhi kriteria berikut :

1 Memungkinkan untuk mencari nilai e, d, n sedemikian rupa sehingga Med mod n = M untuk semua M < n.

2 Relatif mudah untuk menghitung nilai Me mod n dan Cd mod n untuk semua nilai M < n.

3 Tidak memungkinkan mencari nilai d jika diberikan nilai n dan e. Syarat nilai e dan d ini, gcd(d,e)=1

Sebelum memulai penggunaan algoritma RSA ini, terlebih dahulu harus memiliki bahan – bahan dasar sebagai berikut :

1. p dan q = dua bilangan prima yang harus dirahasiakan

2. n, nilai dari hasil p.q

3. e, dengan ketentuan gcd (Φ(n), e) =1

4. d, e-1 (mod Φ(n))

## 2.5 Kompresi Shannon-Fano

Metode pertama yang dikenal untuk mengkodekan simbol secara efektif adalah Shannon Fano. Claude Shannon di Bell Labs dan R M Fano di MIT mengembangkan metode ini secara bersamaan. Pada metode Shannon Fano kompresi dapat dilakukan dengan 2 cara, yaitu dengan membangun pohon biner dan dengan membangun tabel pembagian berdasarkan probabilitas pada setiap simbol. Metode ini dapat dilakukan dengan mengetahui probabilitas dari setiap simbol kemunculan simbol pada sebuah pesan. Dengan mengetahui probabilitas, sebuah tabel kode dapat dibangun dengan properti sebagai berikut:

1. Setiap simbol yang berbeda memiliki kode yang berbeda.

2. Simbol dengan probabilitas kemunculan yang lebih kecil memiliki kode panjang bit yang lebih panjang dan simbol dengan probabilitas yang lebih besar memiliki panjang bit yang lebih pendek.

3. Meskipun kode yang dihasilkan memiliki panjang bit yang berbeda dengan kode pada karakter asli, tetapi dapat didekodekan secara unik (Nelson,1996).

Berikut adalah langkah-langkah dalam Algoritma metode Shannon Fano:

1. Buatlah daftar probabilitas atau daftar kehadiran setiap simbol dari data (pesan yang akan dikodekan).

2. Urutkanlah daftar tersebut menurut frekuensi kemunculan atau kehadiran tiap simbol secara menurun (dari simbol yang frekuensi kemunculannya paling banyak sampai simbol dengan frekuensi kemunculan paling sedikit).

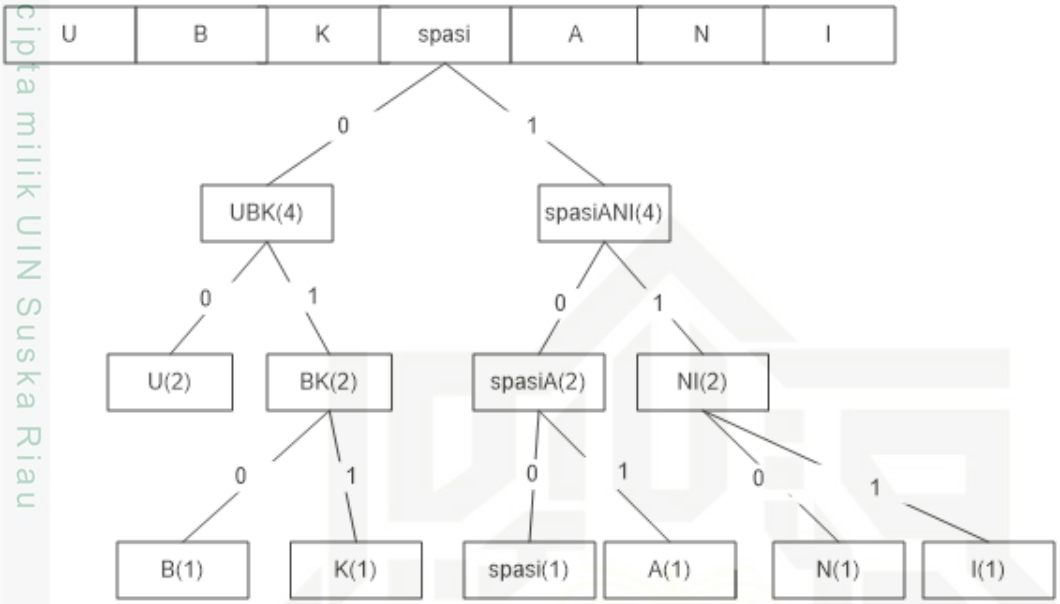
3. Bagilah daftar tersebut menjadi dua bagian dengan pembagian didasari pada jumlah total frekuensi suatu bagian atas sedekat mungkin dengan jumlah frekuensi dengan bagian bawah.

4. Daftar bagian atas dengan digit 0 dan bagian bawah dinyatakan dengan digit 1. Hal tersebut berarti kode untuk simbol-simbol pada bagian atas akan dimulai dengan 0 dan kode untuk simbol-simbol pada bagian bawah akan dimulai dengan 1.

5. Lakukanlah proses secara rekursif langkah 3 dan 4 pada bagian atas dan bawah. Bagilah menjadi kelompok-kelompok dan tambahkanlah bit-bit pada kode sampai setiap simbol memperoleh kode.

Berikut contoh dari penerapan kompresi Shanno-Fano pada penelitian (Christine Lamorahan, Benny Pinontoan, 2013) :

Contoh BUKU ANI




**Gambar 2.4 Pohon Biner Shannon-Fano BUKU ANI**

Untuk melihat ukurannya dapat dilihat pada table yang ada dibawah ini :

**Tabel : 2.2 Pohon Biner Shannon-Fano BUKU ANI**

| Karakter | Biner | Frekuensi | Jumlah |
|---|---|---|---|
| B | 010 | 1 | 1x3 bit = 3 bit |
| U | 00 | 2 | 2x2 bit = 4 bit |
| K | 011 | 1 | 1x3 bit = 3 bit |
| Spasi | 100 | 1 | 1x3 bit = 3 bit |
| A | 101 | 1 | 1x3 bit = 3 bit |
| N | 110 | 1 | 1x3 bit = 3 bit |
| I | 111 | 1 | 1x3 bit = 3 bit |
| | | | Total : 22 bit |

## 2.6 Base64

Base64 adalah salah satu algoritma untuk melakukan penyandian suatu data kedalam format ASCII, yang didasarkan pada bilangan 64 yang digunakan untuk penyandian data biner (Ariyus, 2008). Karakter yang dihasilkan pada transformasi

base64 ini terdiri dari A…Z, a…z dan 0…9 serta ditambah dengan dua karakter terakhir tambah (+) atau garis miring (/) atau sama dengan (=) yang digunakan untuk penyesuaian dan menggenapkan data biner atau istilahnya pengisi *pad.*

Berikut cara transformasi atau konversi base64 :

**Base64 Encoding Table**

| Value | Char | Value | Char | Value | Char | Value | Char |
|---|---|---|---|---|---|---|---|
| 0 | A | 16 | Q | 32 | g | 48 | w |
| 1 | B | 17 | R | 33 | h | 49 | x |
| 2 | C | 18 | S | 34 | i | 50 | y |
| 3 | D | 19 | T | 35 | j | 51 | z |
| 4 | E | 20 | U | 36 | k | 52 | 0 |
| 5 | F | 21 | V | 37 | l | 53 | 1 |
| 6 | G | 22 | W | 38 | m | 54 | 2 |
| 7 | H | 23 | X | 39 | n | 55 | 3 |
| 8 | I | 24 | Y | 40 | o | 56 | 4 |
| 9 | J | 25 | Z | 41 | p | 57 | 5 |
| 10 | K | 26 | a | 42 | q | 58 | 6 |
| 11 | L | 27 | b | 43 | r | 59 | 7 |
| 12 | M | 28 | c | 44 | s | 60 | 8 |
| 13 | N | 29 | d | 45 | t | 61 | 9 |
| 14 | O | 30 | e | 46 | u | 62 | + |
| 15 | P | 31 | f | 47 | v | 63 | / |

**Gambar 2.5 Tabel Base64**

(sumber: https://stackabuse.com/)

Contoh kata Man

M nilai ASCII nya 77, biner nya adalah 01001101

a nilai ASCII nya 97, biner nya adalah 01100001

n nilai ASCII nya 110, biner nya adalah 01101110

Jadikan biner tersebut menjadi 010011010110000101101110, karakter yang akan diproses ada 3, 3 x 8 = 24 bit. 24 bit ini akan diubah menjadi karakter 6 bit, dan mendapatkan 4 karakter. Potong biner menjadi 6 bit.

010011-010110-000101-101110

010011 = 19 = T

010110 = 22 = W

000101 = 5  = F

101110 = 46 = u

Jadi kata Man ketika ditransformasikan kedalam base64 akan menjadi TWFu. Dikarenakan kata Man memiliki 3 karakter atau 24 bit dan habis dibagi 3 maka tidak ada penambahan *padding.*

## 2.7 Penelitian Terkait

Adapun penelitian yang terkait dengan algoritma *rsa dan idea* juga pengamanan data teks dengan android seperti pada penelitian yang dilakukan oleh (Ginting et al., 2015) dengan penelitian implementasi algoritma kriptografi rsa pada e-mail pada penelitian ini menjelaskan tentang penggunaan algoritma rsa untuk melakukan enkripsi dan pengamanan pesan pada email. Pada penelitian ini sudah bagus dan lumayan kuat dalam melakukan enkripsi akan tetapi kunci yang dipakai terlalu kecil hingga membuat kemanannya berkurang.

Pada penelitian selanjutnya oleh (Wulansari et al., 2016) juga yang meneliti mengukur kecepatan enkripsi dan dekripsi algoritma rsa pada pengembangan sistem informasi *text security* , pada penelitian ini membandingkan kecepatan logika algoritma dalam melakukan enkripsi data teks. Pada penelitian hanya melihat kecepatan enkripsi tidak melihat kualitas data yang diperoleh dari enkripsi tersebut.

Pada penelitian lainnya oleh (Wiryadinata, 2007) meneliti tentang kompresi teks pada kompresi shannon-fano. Dengan statis Shannon-fano dan dynamic Shannon-fano. Pada penelitian ini melihat kompleksitas pengujian kompresi dengan melihat kompresi rasio setelah proses kompresi.

Pada penelitian yang dilakukan oleh(Haida Dafitri, Divi Handoko, Imran Lubis, 2010) yang meneliti kompreesi Shannon-fano untuk data teks. Pada penelitian ini mereka meneliti kompleksitas kompresi dengan menguji pengurangan

ukuran data dengan melihat rasio kompresi. Pada penelitian ini hanya melakukan pengompresan data saja.

# BAB III
# METODOLOGI PENELITIAN

Metodologi penelitian ini menjelaskan tentang langkah-langkah dan alur yang akan dilakukan pada penelitian agar penelitian ini dapat berjalan sesuai dengan prosedur dan dapat mencapai tujuan dengan hasil yang baik dan maksimal. Adapun tahapan-tahapan dari penelitian ini dapat dilihat pada gambar 3.1 dibawah ini :


**Gambar 3.1 Metodologi Penelitian**

## 3.1 Identifikasi Masalah

Pada tahap identifikasi masalah ini yaitu mencari dan memahami permasalahan yang ada pada bidang kriptografi. Pada tahap ini harus mengetahui permasalahan yang ada pada penelitian-penelitian sebelumnya yang terkait tentang kriptografi bagian pengamanan data teks dan implementasinya kedalam kehidupan sehari-hari.

Pada penelitian sebelumnya kriptografi pada pengamanan data text sudah ada dalam beberapa metode atau algoritma yang diangkat dan diimplementasikan seperti metode atau algoritma RSA . Menurut penelitian yang menyangkut tentang metode atau algoritma RSA, algoritma ini memiliki beberapa keunggulan dibandingkan dengan metode lain diantaranya dikarenakan algoritma ini memiliki dua kunci yaitu *public key* dan *private key*.Pada penelitian yang meneliti kompresi menggunakan Shannon-Fano, kompresi ini memiliki keunggulan dibandingkan dengan algoritma kompresi yang lainnya. Kompresi ini memiliki keunggulan dengan tingkat rasio kompresi yang tinggi.

Maka dari itu pada penelitian ini akan meneliti tentang enkripsi dan dekripsi algoritma RSA pada file text. Pada penelitian ini data text akan di enkripsi dan dekripsi dengan algoritma RSA dan setelah itu dikompress dengan menggunakan algoritma Shannon – Fano untuk melihat perubahan data teks yang ada dari segi ukuran, kecepatan dan perubahan data yang ada sebelum dikompress maupun setelah dikompress .

## 3.2 Pengumpulan Data

Pada tahap ini yaitu pengumpulan data serta informasi dari berbagai sumber yang berkaitan dengan penelitian yaitu implementasi enkripsi dan enkripsi algoritma RSA dan kompresi Shannon-Fano pada data text dan teori enkripsi serta kompresi.

Adapun metoda yang dilakukan adalah sebagai berikut :

a. Studi Literatur

   Pada metoda studi literature ini berupa pencarian serta mendapatkan informasi dari berbagai referensi seperti jurnal, buku serta penelitian

yang sudah dilakukan sebelumnya. Data dan informasi yang dikumpulkan didapat dari *Google Scholar*, Portal Garuda, dan *Researchgate*

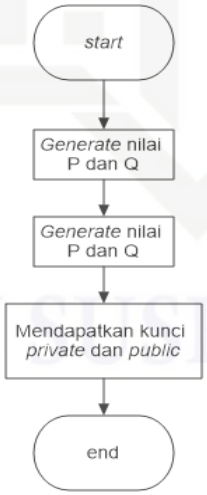
b. Diskusi

Pada metoda ini dilakukan diskusi dalam berbagai forum serta komunitas yang menyangkut tentang pemrograman desktop atau web serta kriptografi untuk mencari solusi tentang permasalahan yang berhubungan dengan kesulitan pemrograman dan juga kriptografi.

## 3.3 Analisa dan Perancangan

Pada tahap ini akan dilakukan analisa sistem dan analisa desain tampilan sistem. Pada tahap analisa akan dilakukan analisa terhaap permasalahan, analisa sistem input maupun output. Menganalisa cara kerja algoritma RSA dalam menentukan *public key* dan *private key*, melakukan enkripsi dan dekripsi file text dan juga menganalisa cara kerja kompresi Shannon -Fano dalam mengompressi file text setelah di enkripsi. Analisa dan alur kompresi serta enkripsi bias dilihat dibawah ini:

### 3.3.1 Proses pembangkitan kunci



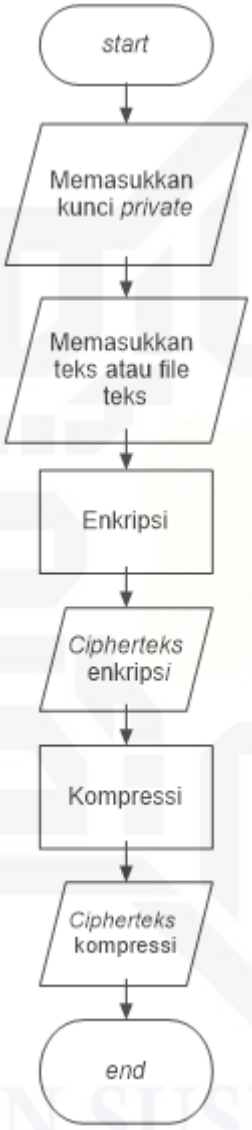

**Gambar 3.2 Flowchart Pembangkitan Kunci**

Penjelasan :

a. Melakukan generate bilangan p dan q prima

b. Mendapatkan modulus / N dari p x q
c. Mendapatkan kunci *public* dan *private*
d. selesai

3.3.2 Proses pengirim enkripsi dan kompresi



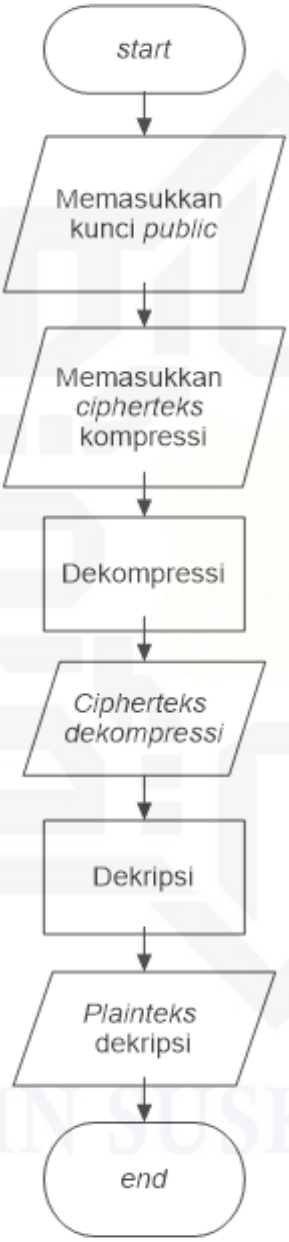

**Gambar 3.3 Flowchart Pengirim**

Penjelasan :

a. Memasukkan kunci *private* yag sudah dibangkitkan
b. Memasukkan teks atau file yang berejenis teks / txt
c. Melakukan enkripsi
d. Mendapatkan *cipherteks* hasil enkripsi

e. Melakukan kompresi
f. Mendapatkan *cipherteks* hasil kompresi
g. Selesai

3.3.3 Proses penerima dekompresi dan dekripsi




**Gambar 3.4 Flowchart Penerima**

Penjelasan :

a. Memasukkan kunci *public*
b. Memasukkan teks atau file txt yang sudah dikompress

c. Melakukan dekompresi
d. Mendapatkan *cipherteks* hasil dekompresi
e. Melakukan dekripsi
f. Mendapatkan *plainteks* awal
g. selesai

Pada tahap perancangan akan dilakukan rancangan sistem yang akan dibuat, seperti rancangan tampilan sistem yang didapat dari tahap analisa sebelumnya. Dalam perancangan inilah merancang sistem yang akan dibangun berdasarkan analisa dan cara kerja algoritma yang telah ditentukan

## 3.4 Implementasi dan Pengujian

Implementasi dan pengujian ini menjelaskan tentang bagaimana mengimplementasikan sistem yang telah dibangun berdasarkan analisa dan perancangan sebelumnya. Sistem yang telah dibangun akan diuji coba untuk mengetahui kelemahan atau kekurangan dan keberhasilan sistem yang telah dibangun sesuai dengan yang telah ditentukan atau ditargetkan sebelum membangun sistem.

Pengujian akan dilakukan dengan cara sebagai berikut :

a. Pengujian kecepatan yang dilakukan dengan melihat kecepatan tiap proses yang ada
b. Pengujian kapasitas ukuran yang dilakukan dengan melihat ukuran data enkripsi, kompresi, dekompresi dan dekripsi
c. Pengujian *recovery* yang dilakukan dengan melihat apakah tiap proses berjalan sesuai dengan yang diharapkan atau sebaliknya
d. Pengujian ketahanan, dilakukan penyerangan dengan menggunakan metode *brute force attack* untuk melihat kekuatan algoritma

## 3.5 Kesimpulan dan Saran

Dalam tahap ini merumuskan kesimpulan-kesimpulan dalam penelitian Implementasi Enkripsi dan Dekripsi Menggunakan Algoritma RSA dan Kompresi Sahnnon - Fano Untuk Keamanan Data text. Serta saran dalam pengembangan dan penyempurnaan dari penelitian ini.

# BAB IV
# ANALISA DAN PERANCANGAN

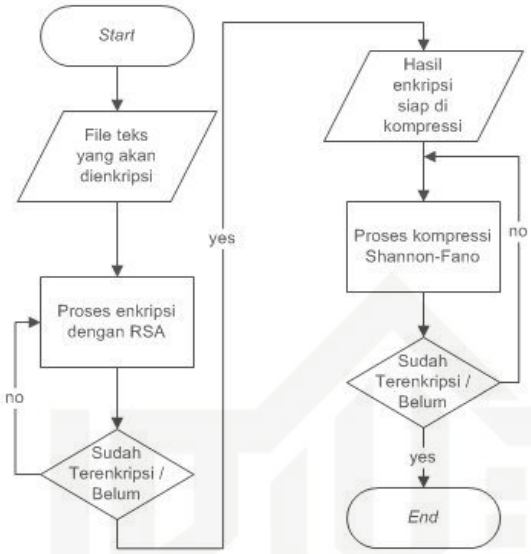
## 4.1 Analisa dan Deskripsi Umum

Pada dasarnya kriptografi diciptakan untuk melakukan pengamanan pesan dari pihak ketiga yang ingin mengetahui isi pesan yang ingin dirahasiakan. Kriptografi yang diangkat adalah kriptografi RSA dan kompresi Shannon-Fano. Pada penelitian ini algoritma RSA akan digunakan dalam pengamanan data teks melakukan enkripsi dan dekripsi dengan menggunakan dua buah kunci. Untuk efisiensi ruang atau mengecilkan data hasil enkripsi dan dekripsi maka diakukan kompresi dengan menggunakan algoritma Shannon-Fano.

Kriptografi RSA ini merupakan kriptografi asimetris yang memiliki dua buah kunci yaitu kunci *Private* dan kunci *Public.* Berikut langkah pembangkitan kunci :

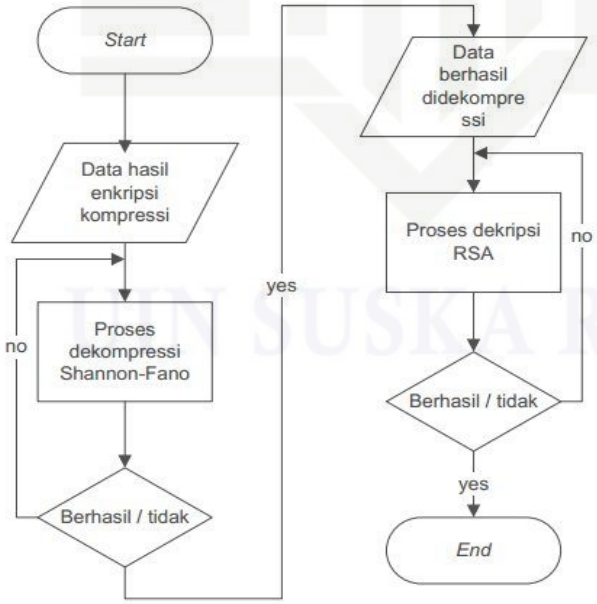
1. Menentukan nilai P dan Q yang prima, dan disini diberi rentang nilai 0-1.000.000
2. Setelah mendapatkan nilai P dan Q maka akan menentukan nilai Modulus / N dengan menggunakan rumus N = P x Q
3. Setelah mendapat nilai n maka mencari nilai totient
4. Kunci *public* dan kunci *private* sudah didapatkan

Setelah mendapatkan kunci *private* dan kunci *public* maka melakukan pertukaran kunci antara penerima dan pengirim pesan, setelah itu maka dilakukan enkripsi pesan. Pesan yang akan dienkripsi berupa teks atau file yang berjenis teks. Untuk proses enkripsi-kompresi dapat dilihat pada Gambar 4.1 berikut :




**Gambar 4.1 Flowchart Enkripsi - Kompresi**

Setelah teks yang di enkripsi sebelumnya dikompress maka akan menghasilkan cipherteks baru dan ukuran data yang berbeda dari sebelum di enkripsi dan kompress. Untuk mengubahnya kembali menjadi plainteks awal maka dilakukan proses dekompresi dan dekripsi. Secara umum proses dekompresi dan dekripsi dapat dilihat pada Gambar 4.2 berikut ini:




**Gambar 4.2 Flowchart Dekompresi – Dekripsi**

## 4.2 Analisa Aplikasi

Analisa yang digunakan adalah analisa pengguna, analisa fungsional dan analisa fitur yang ada pada sistem yang akan dibangun.

### 4.2.1 Analisa Pengguna (User)

Pengguna atau *User* yang berada dalam lingkup sistem adalah umum, karena sistem berfungsi untuk melakukan enkripsi teks dan mengkompress nya maka pengguna biasa saja bisa memakai sistem.

### 4.2.2 Analisa Fitur

Pada analisa fitur yang akan dikembangkan pada sistem ini :

1. Pembangkitan Kunci : Fitur ini untuk melakukan pembangkitan kunci RSA dengan menggunakan dua bilangan prima yaitu P dan Q secara random dengan rentang 0 -1.000.000
2. Enkripsi : Fitur ini berfungsi untuk melakukan enkripsi teks
3. Kompresi : Fitur ini berfungsi untuk melakukan kompresi teks
4. Dekompresi : Fitur ini berfungsi untuk melakukan dekompresi teks yang sudah dikompress
5. Dekripsi : Fitur ini berfungsi untuk melakukan dekripsi teks yang telah di enkripsi sebelumnya
6. Ukuran : Fitur ini untuk melihat ukuran teks setelah dienkripsi, kompresi, dekompresi dan dekripsi
7. Kecepatan : Fitur ini untuk melihat kecepatan dalam melakukan pembangkita kunci *private* dan kunci *public*, enkripsi, kompresi, dekompresi dan dekripsi teks

## 4.3 Analisa Perhitungan Manual RSA dan Shannon-Fano

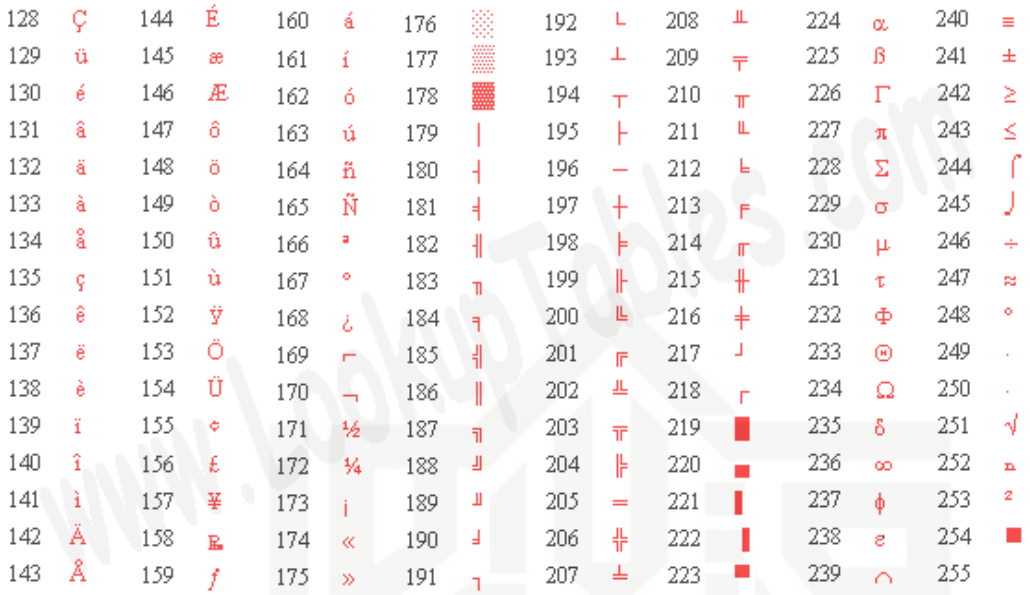
Pada analisa perhitungan manual algoritma RSA dan Shannon-Fano ini akan menjelaskan bagaimana perhitungan algoritma RSA dan Shannon-Fano secara spesifik. Mulai dari pembangkitan kunci, proses enkripsi, proses kompresi, proses dekompresi dan proses dekripsi. Sebelum melakukan perhitungan manual RSA dan

Shannon-Fano, memerlukan panduan tabel ASCII untuk panduan hasil dari perhitungan manual RSA dan Shannon-Fano.

Kode ASCII merupakan kode standar Amerika dalam melakukan pertukaran informasi. Kode ASCII merupakan standar internasional dalam bentuk huruf dan symbol, tetapi ASCII bersifat universal. Kode ASCII sebenarnya hanya memliki 7 bit biner akan tetapi ASCII disimpan sebagai 8 bit sebagai nilai bit tertinggi, untuk bilangan ASCII dapat dilihat pada gambar dibawah ini :

| Dec | Hx | Oct | Char | Dec | Hx | Oct | Html | Chr | Dec | Hx | Oct | Html | Chr | Dec | Hx | Oct | Html | Chr |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 000 | NUL (null) | 32 | 20 | 040 |   | Space | 64 | 40 | 100 | @ | @ | 96 | 60 | 140 | ` | ` |
| 1 | 1 | 001 | SOH (start of heading) | 33 | 21 | 041 | ! | ! | 65 | 41 | 101 | A | A | 97 | 61 | 141 | a | a |
| 2 | 2 | 002 | STX (start of text) | 34 | 22 | 042 | " | " | 66 | 42 | 102 | B | B | 98 | 62 | 142 | b | b |
| 3 | 3 | 003 | ETX (end of text) | 35 | 23 | 043 | # | # | 67 | 43 | 103 | C | C | 99 | 63 | 143 | c | c |
| 4 | 4 | 004 | EOT (end of transmission) | 36 | 24 | 044 | $ | $ | 68 | 44 | 104 | D | D | 100 | 64 | 144 | d | d |
| 5 | 5 | 005 | ENQ (enquiry) | 37 | 25 | 045 | % | % | 69 | 45 | 105 | E | E | 101 | 65 | 145 | e | e |
| 6 | 6 | 006 | ACK (acknowledge) | 38 | 26 | 046 | & | & | 70 | 46 | 106 | F | F | 102 | 66 | 146 | f | f |
| 7 | 7 | 007 | BEL (bell) | 39 | 27 | 047 | ' | ' | 71 | 47 | 107 | G | G | 103 | 67 | 147 | g | g |
| 8 | 8 | 010 | BS (backspace) | 40 | 28 | 050 | ( | ( | 72 | 48 | 110 | H | H | 104 | 68 | 150 | h | h |
| 9 | 9 | 011 | TAB (horizontal tab) | 41 | 29 | 051 | ) | ) | 73 | 49 | 111 | I | I | 105 | 69 | 151 | i | i |
| 10 | A | 012 | LF (NL line feed, new line) | 42 | 2A | 052 | * | * | 74 | 4A | 112 | J | J | 106 | 6A | 152 | j | j |
| 11 | B | 013 | VT (vertical tab) | 43 | 2B | 053 | + | + | 75 | 4B | 113 | K | K | 107 | 6B | 153 | k | k |
| 12 | C | 014 | FF (NP form feed, new page) | 44 | 2C | 054 | , | , | 76 | 4C | 114 | L | L | 108 | 6C | 154 | l | l |
| 13 | D | 015 | CR (carriage return) | 45 | 2D | 055 | - | - | 77 | 4D | 115 | M | M | 109 | 6D | 155 | m | m |
| 14 | E | 016 | SO (shift out) | 46 | 2E | 056 | . | . | 78 | 4E | 116 | N | N | 110 | 6E | 156 | n | n |
| 15 | F | 017 | SI (shift in) | 47 | 2F | 057 | / | / | 79 | 4F | 117 | O | O | 111 | 6F | 157 | o | o |
| 16 | 10 | 020 | DLE (data link escape) | 48 | 30 | 060 | 0 | 0 | 80 | 50 | 120 | P | P | 112 | 70 | 160 | p | p |
| 17 | 11 | 021 | DC1 (device control 1) | 49 | 31 | 061 | 1 | 1 | 81 | 51 | 121 | Q | Q | 113 | 71 | 161 | q | q |
| 18 | 12 | 022 | DC2 (device control 2) | 50 | 32 | 062 | 2 | 2 | 82 | 52 | 122 | R | R | 114 | 72 | 162 | r | r |
| 19 | 13 | 023 | DC3 (device control 3) | 51 | 33 | 063 | 3 | 3 | 83 | 53 | 123 | S | S | 115 | 73 | 163 | s | s |
| 20 | 14 | 024 | DC4 (device control 4) | 52 | 34 | 064 | 4 | 4 | 84 | 54 | 124 | T | T | 116 | 74 | 164 | t | t |
| 21 | 15 | 025 | NAK (negative acknowledge) | 53 | 35 | 065 | 5 | 5 | 85 | 55 | 125 | U | U | 117 | 75 | 165 | u | u |
| 22 | 16 | 026 | SYN (synchronous idle) | 54 | 36 | 066 | 6 | 6 | 86 | 56 | 126 | V | V | 118 | 76 | 166 | v | v |
| 23 | 17 | 027 | ETB (end of trans. block) | 55 | 37 | 067 | 7 | 7 | 87 | 57 | 127 | W | W | 119 | 77 | 167 | w | w |
| 24 | 18 | 030 | CAN (cancel) | 56 | 38 | 070 | 8 | 8 | 88 | 58 | 130 | X | X | 120 | 78 | 170 | x | x |
| 25 | 19 | 031 | EM (end of medium) | 57 | 39 | 071 | 9 | 9 | 89 | 59 | 131 | Y | Y | 121 | 79 | 171 | y | y |
| 26 | 1A | 032 | SUB (substitute) | 58 | 3A | 072 | : | : | 90 | 5A | 132 | Z | Z | 122 | 7A | 172 | z | z |
| 27 | 1B | 033 | ESC (escape) | 59 | 3B | 073 | ; | ; | 91 | 5B | 133 | [ | [ | 123 | 7B | 173 | { | { |
| 28 | 1C | 034 | FS (file separator) | 60 | 3C | 074 | < | < | 92 | 5C | 134 | \ | \ | 124 | 7C | 174 | | | \| |
| 29 | 1D | 035 | GS (group separator) | 61 | 3D | 075 | = | = | 93 | 5D | 135 | ] | ] | 125 | 7D | 175 | } | } |
| 30 | 1E | 036 | RS (record separator) | 62 | 3E | 076 | > | > | 94 | 5E | 136 | ^ | ^ | 126 | 7E | 176 | ~ | ~ |
| 31 | 1F | 037 | US (unit separator) | 63 | 3F | 077 | ? | ? | 95 | 5F | 137 | _ | _ | 127 | 7F | 177 | &#127; | DEL |

**Gambar 4.3 Tabel ASCII** ( Sumber: http://www.asciitable.com/ )



**Gambar 4.4 Tambahan Tabel ASCII** (Sumber: http://www.asciitable.com/)

Untuk perhitungan manual RSA dan Shannon-Fano dapat dilihat dibawah ini :

### 4.4.1 Perhitungan RSA

RSA merupakan algoritma asimetris yang memakai dua buah kunci dalam melakukan enkripsi dan dekripsi nya. Ini merupakan contoh perhitungan manual dari algoritma RSA itu sendiri :

Plainteks : ADIDAYA

**Tabel 4.2 Plainteks**

| Karakter | A | D | I | D | A | Y | A |
|---|---|---|---|---|---|---|---|
| Decimal | 65 | 68 | 73 | 68 | 65 | 89 | 65 |

1. Menentukan nilai p dan q yang mana nila p dan q harus bernilai prima

   P = 23

   Q = 19

2. Mencari nilai n

   N = p x q

   N= 23 x 19

N= 437

3. Mencari nilai totient

   Φ = (p-1)(q-1)

   Φ = (23-1)(19-1)

   Φ = (22)(18)

   Φ = 396

4. Mencari nilai e dengan persamaan GCD (e, Φ)=1

   396 mod 7 = 4

   7 mod 4 = 3

   4 mod 3 = 1

   Jadi nilai e nya adalah 7

5. Mencari nilai d dengan persamaan d (d,e) mod Φ = 1

   (d,e) mod Φ = 1

   (283x7) mod 396 =1

   1981 mod 396 = 1

   Jadi nilai d nya adalah 283

Jadi didapatkan nilai dari masing masing variable :

P = 23

Q = 19

N = 437

Φ = 396

E = 7

D = 283

Jadi kunci *private* nya adalah : n = 437 d= 283

Dan kunci *public* nya adalah : n= 437 e = 7

Plainteks yang akan di enkripsi adalah ADIDAYA

Enkripsi

Rumus enkripsi RSA yang merujuk pada landasan teori RSA adalah :

$$C = Me \bmod n \ (Fungsi\ Enkripsi)$$

**Tabel 4.3 Enkripsi RSA**

| Karakter | Perhitungan |
|---|---|
| A | $= 65^7 \bmod 437$<br>$= 84$ |
| D | $= 68^7 \bmod 437$<br>$= 68$ |
| I | $= 73^7 \bmod 437$<br>$= 169$ |
| D | $= 68^7 \bmod 437$<br>$= 68$ |
| A | $= 65^7 \bmod 437$<br>$= 84$ |
| Y | $= 89^7 \bmod 437$<br>$= 67$ |
| A | $= 65^7 \bmod 437$<br>$= 84$ |

**Tabel 4.4 Hasil Enkripsi**

| Decimal | 84 | 68 | 169 | 68 | 84 | 67 | 84 |
|---|---|---|---|---|---|---|---|
| Karakter | T | D | © | D | T | C | T |

Setelah di enkripsi maka plainteks ADIDAYA berubah menjadi TD©DTCT dalam karakter dan untuk keamanan cipherteks TD©DTCT akan dikonversi kedalam bentuk base64.

**Tabel 4.5 Tabel Biner Enkripsi**

| Decimal | 84 | 68 | 169 | 68 | 84 | 67 | 84 |
|---|---|---|---|---|---|---|---|
| Karakter | T | D | © | D | T | C | T |
| Biner | 01010100 | 01000100 | 10101001 | 01000100 | 01010100 | 01000011 | 01010100 |

Biner 8 bit dari cipherteks TD©DTCT akan dijadikan biner Panjang secara keseluruhan 010101000100010010101001010001000101010001000011010101000, dalam base 64 biner 8 bit akan dijadikan biner 6 bit. Setelah dijadikan biner bit akan menjadi 010101-000100-010010-101001-010001-000101-010001-000011-

010101-00. Untuk yang bukan kelipatan 3 akan ada *padding* = yang akan menambah Panjang kata mencapai kelipatan 3 terdekat.

**Tabel 4.6 Hasil Base64**

| Biner | Desimal | Karakter Base64 |
|---|---|---|
| 00010101 | 21 | V |
| 00000100 | 4 | E |
| 00010010 | 18 | S |
| 00101001 | 41 | p |
| 00010001 | 17 | R |
| 00000101 | 5 | F |
| 00010001 | 17 | R |
| 00000011 | 3 | D |
| 00010101 | 21 | V |
| 00000000 | 0 | A |

Maka cipherteks TD©DTCT akan menjadi VESpRFRDVA== dalam base64.

Dekripsi

Sebelum melakukan dekripsi yang pertama dilakukan adalah melakukan konversi base64 ke ASCII terlebih dahulu. *Cipherteks* VESpRFRDVA== akan dikonversi kebentuk ASCII. Menjadikan *cipherteks* base64 kedalam bentuk biner 6 bit

010101-000100-010010-101001-010001-000101-010001-000011-010101-00

Menjadikan biner 6 bit menjadi biner 8 bit

01010100-01000100-10101001-01000100-01010100-01000011-01010100

Setelah menjadi biner 8bit maka akan *cipherteks* VESpRFRDVA== akan menjadi TD©DTCT dalam ASCII bias dilihat pada table 4. .

Rumus dekripsi RSA yang merujuk pada landasan teori RSA adalah :

$$M = Cd \bmod n \ (Fungsi\ Dekripsi)$$

**Tabel 4.7 Dekripsi RSA**

| Karakter | Perhitungan |
|---|---|
| T | = $84^{283}$ mod 437<br>= 65 |
| D | = $68^{283}$ mod 437<br>= 68 |
| © | = $169^{283}$ mod 437<br>= 73 |
| D | = $68^{283}$ mod 437<br>= 68 |
| T | = $84^{283}$ mod 437<br>= 65 |
| C | = $67^{283}$ mod 437<br>= 89 |
| T | = $84^{283}$ mod 437<br>= 65 |

**Tabel 4.8 Hasil Dekripsi RSA**

| Decimal | 65 | 68 | 73 | 68 | 65 | 89 | 65 |
|---|---|---|---|---|---|---|---|
| Karakter | A | D | I | D | A | Y | A |

Setelah dilakukan dekripsi cipherteks TD©DTCT menjadi plainteks ADIDAYA dalam karakter.

### 4.4.2 Perhitungan Shannon-Fano

Shannon-Fano merupakan algoritma untuk melakukan kompresi. Pada umumnya Shannon-Fano hampir sama dengan algoritma Huffman dalam melakukan kompresi, akan tetapi algoritma Shannon-Fano lebih efektif untuk melakukan kompresi data yang panjang. Ini merupakan contoh dari perhitungan kompresi Shannon-Fano itu sendiri :

Kompresi

Cipherteks hasil enkripsi RSAdapat dilihat pada tabel dibawah :

**Tabel 4.9 Hasil Enkripsi RSA Yang Akan Dikompresi**

| Karakter | T | D | © | D | T | C | T |
|---|---|---|---|---|---|---|---|
| Decimal | 84 | 68 | 169 | 68 | 84 | 67 | 84 |
| Base64 | VESpRFRDVA== | | | | | | |

Untuk melihat ukuran teks sebelum dilakukan kompresi dapat dilihat pada tabel dibawah ini :

**Tabel 4.10 Ukuran Teks Sebelum Dikompress**

| Biner | Karakter | Ukuran |
|---|---|---|
| 01010110 | V | 8 bit |
| 01000101 | E | 8 bit |
| 01010011 | s | 8 bit |
| 01110000 | P | 8 bit |
| 01010010 | R | 8 bit |
| 01000110 | F | 8 bit |
| 01010010 | R | 8 bit |
| 01000100 | D | 8 bit |
| 01010110 | V | 8 bit |
| 01000001 | A | 8 bit |
| 00111101 | = | 8 bit |
| 00111101 | = | 8 bit |
| | Total Ukuran | 96 bit |

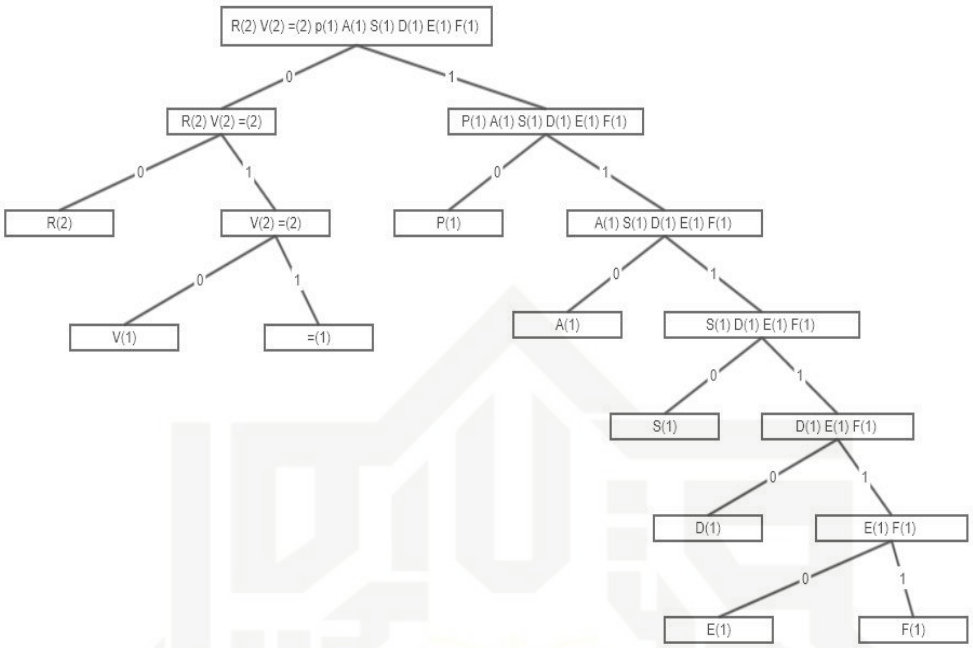
Pada tabel diatas dapat dilihat jadi total ukuran cipherteks sebelum dikompress adalah 56 bit atau 7 byte. Berikut tahapan dalam melakukan kompresi:

1. Hitung frekuensi kemunculan tiap karakter :

**Tabel 4.11 Hitung Frekuensi Kemunculan Karakter**

| Karakter | R | V | = | p | A | S | D | E | F |
|---|---|---|---|---|---|---|---|---|---|
| Jumlah | 2 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 |
| Probabilitas | 0,17 | 0,17 | 0,17 | 0,08 | 0,08 | 0,08 | 0,08 | 0,08 | 0,08 |

2. Membangun pohon biner tiap karakter berdasarkan frekuensi terbesar dengan sama rata



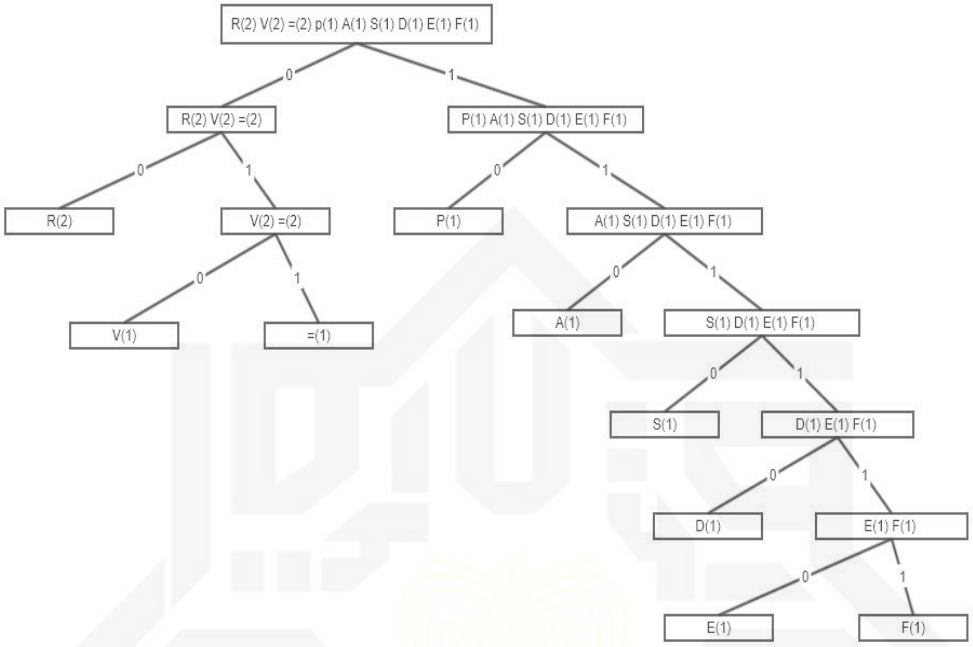

**Gambar 4.5 Pohon Biner Kompresi**

**Tabel 4.12 Pohon Biner Shannon Fano**

| Karakter | Jumlah | Bit Kode |
|---|---|---|
| R | 2 | 00 |
| V | 2 | 010 |
| = | 2 | 011 |
| p | 1 | 10 |
| A | 1 | 110 |
| S | 1 | 1110 |
| D | 1 | 11110 |
| E | 1 | 111110 |
| F | 1 | 111111 |

Setelah membuat pohon biner dan mendapatkan bit kode pohon biner Shannon-fano maka dapat dilakukan substitusi dari *cipherteks* VESpRFRDVA== sehingga diperoleh data biner sebagai berikut 010111110111010001111110011110010110011011. Data biner hasil substitusi Shannon-fano memiliki ukuran 42 bit. Ukuran setelah dikompresi menjadi 96 bit – 42 bit = 54 bit

Dekompresi




**Gambar 4.6 Pohon Biner Dekompresi**

Langkah langkah dekompresi :

1. Membaca bit pertama dari string dari biner masukan
2. Membaca pohon biner jika kekiri maka 0 dan jika kekanan maka 1
3. Jika anak dari pohon bukan daun maka baca bit berikutnya
4. Pada daun tersebut maka karakter akan ditemukan
5. Proses penguraian ini akan dilakukan hingga keseluruhan string masukan diproses

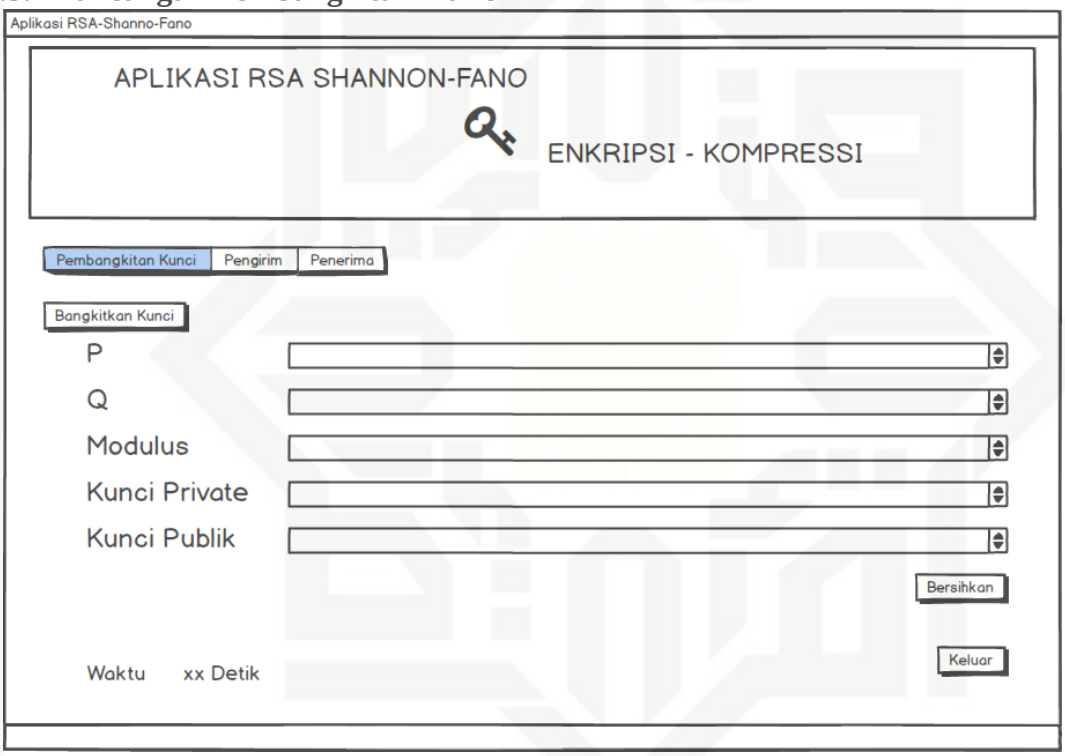
Hasil pengkodean string *cipherteks* dari VESpRFRDVA== kedalam biner adalah 010 111110 1110 10 00 111111 00 11110 010 110 011 011. Bit pertama dari sting tersebut adalah 0 dan diketahui bit sebelah kirinya bukan daun maka akan ditelusuri lagi bit keduanya adalah 1 dan 1 bukan daun, selanjutnya ditelusuri lagi bit ketiga yaitu 0 maka ditemukan daun dan karakter V, dan dilakukan berulang hingga bit terakhir maka ditemukan 0101111101110100011111100111100101110

011011 adalah hasil kompresi dari string atau *cipherteks* dari VESpRFRDVA==.

## 4.4 Perancangan User Interface

Dalam perancangan User Interface ini akan merancang struktur atau tampilan dari sistem atau aplikasi yang dibangun. Berikut merupakan perancangan user interface yang sudah dibuat:

### 4.5.1 Rancangan Pembangkitan Kunci

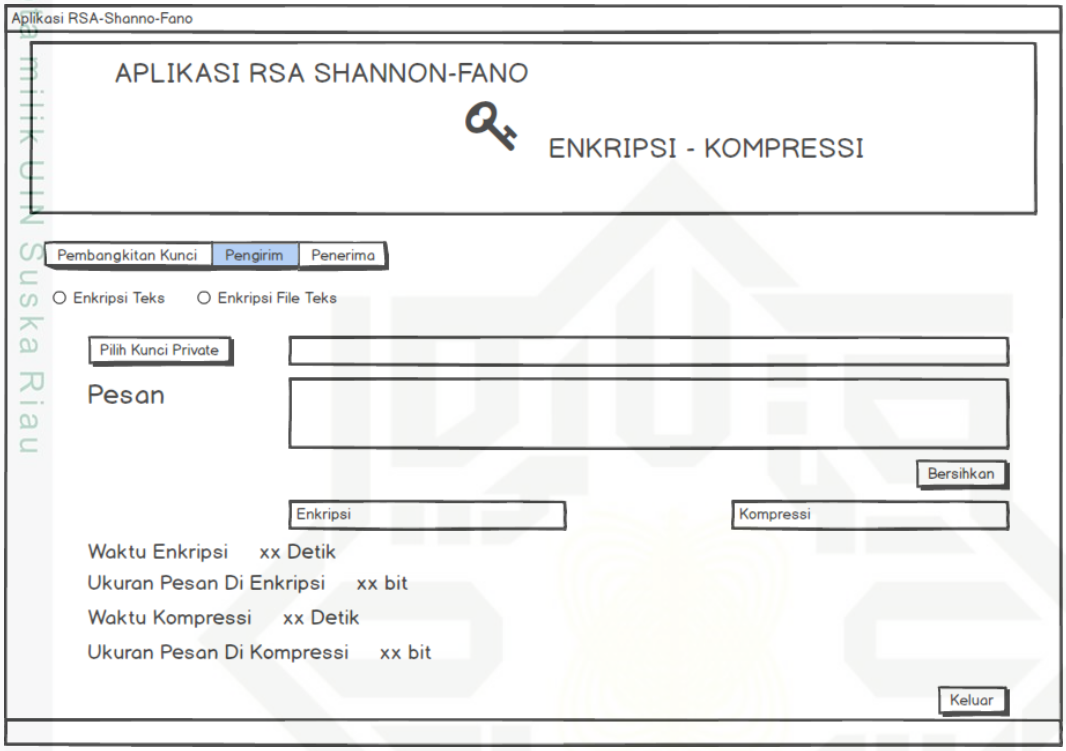



**Gambar 4.7 Rancangan Tampilan Pembangkitan Kunci**

Rancangan tampilan aplikasi ini adalah tampilan yang akan dibuat atau diterapkan pada aplikasi yang dibangun. Pada rancangan tampilan aplikasi ini terdapat beberapa menu yaitu pembangkitan kunci, enkripsi, kompresi, dekompresi, dekripsi dan juga dapat melihat ukuran teks yang asli, teks yang sudah di enkripsi dan juga teks yang sudah dikompresi.

Pada menu pembangkitan kunci, kunci didapat dengan bilangan prima P dan bilangan prima Q yang diambil secara acak didalam aplikasi dan setelah bilangan

prima P dan bilangan prima Q didapat maka didapat modulusnys, setelah itu baru akan didapatkan kunci *public* dan kunci *private* nya. Rancangan Pengirim

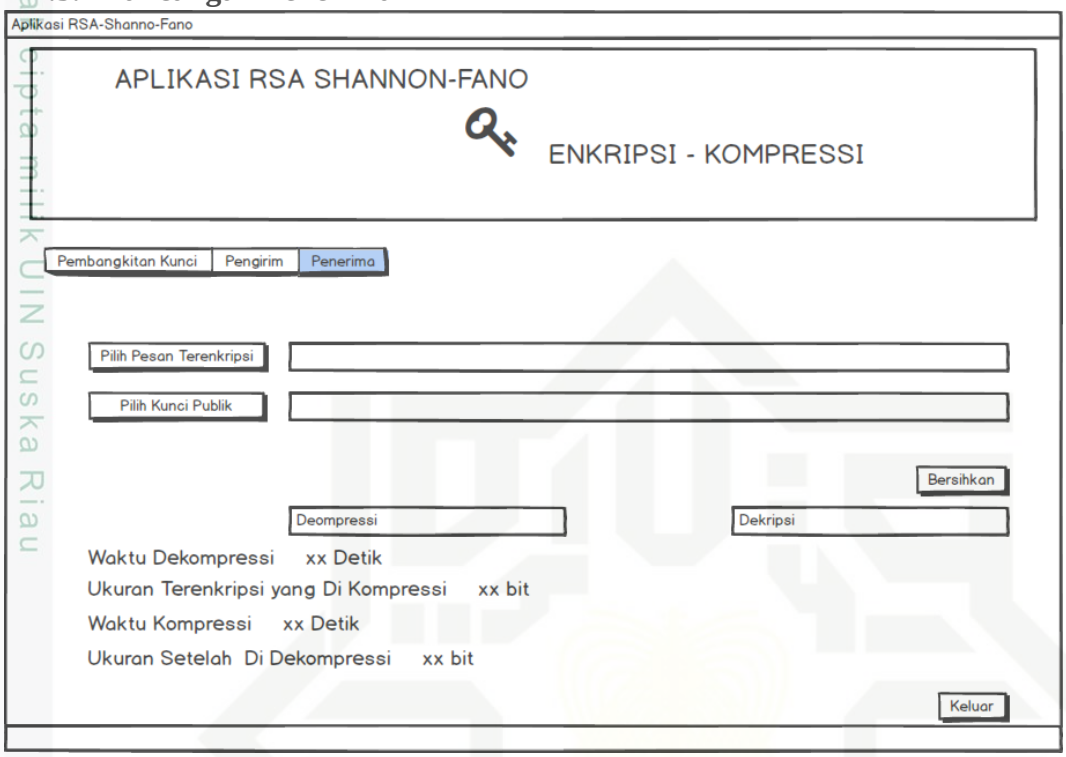



**Gambar 4.8 Rancangan Tampilan Pengirim**

Pada menu enkripsi dapat dilihat pada rancangan tampilan yang dibuat. Sebelum melakukan enkripsi pengguna dapat memasukkan teks ataupun file yang berjenis teks, setelah pengguna memasukkan teks atau file teks pengguna dapat melakukan enkripsi dengan menekan button enkripsi yang ada pada rancangan aplikasi. Hasil dari enkripsi dimasukkan dalam format txt.

Pada menu kompresi dapat dilihat pada rancangan tampilan yang dibuat. Sebelum melakukan kompresi pengguna dapat melakukan kompresi dengan menekan button kompresi, secara otomatis aplikasi akan mengambil hasil enkripsi yang berjenis txt didalam folder untuk dilakukan kompresi. Hasil kompresi dimasukkan kedalam format txt.

### 4.5.2 Rancangan Penerima




**Gambar 4.9 Rancangan Tampilan Penerima**

Pada menu dekompresi dapat dilihat pada rancangan tampilan yang dibuat. Sebelum melakukan dekompresi pengguna dapat melakukan dekompresi dengan menekan button dekompresi, secara otomatis aplikasi akan mengambil hasil kompresi yang berjenis txt didalam folder untuk dilakukan dekompresi. Hasil dekompresi dimasukkan kedalam format txt.

Pada menu dekripsi dapat dilihat pada rancangan tampilan yang dibuat. Sebelum melakukan dekripsi pengguna dapat melakukan dekripsi dengan menekan button dekripsi , secara otomatis aplikasi akan mengambil hasil dekompresi yang berjenis txt didalam folder untuk dilakukan dekripsi. Hasil dekripsi dimasukkan kedalam format txt.

Pada rancangan tampilan yang dibuat juga memuat tanpilan untuk melihat kecepatan tiap proses dari pembangkitan kunci, enkripsi, kompresi, dekompresi dan dekripsi. Dan juga pada rancangan tampilan yang dibuat juga memuat tampilan ukuran teks dari teks asli , teks setelah di enkripsi dan teks setelah di kompresi.

# BAB VI
# PENUTUP

## 6.1 Kesimpulan

Berdasarkan penelitian dari implementasi algoritma RSA dan kompresi Shanno-Fano yang telah dirancang dan analisa sebelumnya dapat diambil kesimpulan sebagai berikut :

1. Dengan memberikan rentang standar 1 – 1.000.000 pada nilai P dan Q dalam pembangkitan kunci dapat mempengaruhi kecepatan dalam melakukan pembangkitan kunci, enkripsi, kompresi, dekompresi dan dekripsi data teks dan file teks.
2. Data teks atau file teks yang sudah dienkripsi menggunakan kunci *public* tidak bisa dibuka menggunakan metode *brute force attack.* Cipherteks yang sudah di enkripsi hanya dapat dibuka menggunakan pasangan kunci *private* yang sudah dibangkitkan sebelumnya.

## 6.2 Saran

Berdasarkan hasil dari implementasi algoritma RSA dan kompresi Shanno-Fano maka penulis menyarankan atau merekomendasikan bebrapa hal sebegai berikut :

1. Untuk meningkatkan kinerja aplikasi kedepannya maka dapat dilakukan dengan menambahkan beberapa jenis file seperti gambar dengan format yang berbeda.
2. Untuk mencakup penggunaan yang lebih besar kedepannya maka dapat dilakukan dengan menambahkan beberapa jenis platform yang berbeda dalam mengaplikasikannya.

# DAFTAR PUSTAKA

Ariyus, D. (2008). *Pengantar Ilmu Kriptografi dan Implementasi*. Yogyakarta: Penerbit Andi.

Asmoro, P. S. (2015). *Pengamanan Data Citra DENGAN Gabungan Algoritma RSA Dan OTP*.

Christine Lamorahan, Benny Pinontoan, N. N. (2013). *Data Compression Using Shannon-Fano Algorithm*.

Ginting, A., Isnanto, R. R., & Windasari, I. P. (2015). *Implementasi Algoritma Kriptografi RSA untuk Enkripsi dan Dekripsi E-Mail*. *3*, 253–258.

Haida Dafitri, Divi Handoko, Imran Lubis, M. N. (2010). *Perancangan Aplikasi Kompresi File Teks Menggunakan Dynamic Method Of Shannon-Fano Algorithm*. 249–254.

Infotama, M. (2010). *Pretty Good Privacy (PGP) (Enskripsi Informasi Dengan Metode Kriptografi)*. *5*(1), 7–18.

Maryanto, Anik Muslikah Indriastuti, D. W. (2014). *Bahasa Indonesia Ekspresi Diri dan Akademik (Buku Siswa) Kelas 11 SMA/SMK*. Pusat Kurikulum dan Perbukuan Kementerian Pendidikan dan Kebudayaan.

Munir, R. (2006). *Kriptografi*. Pasar Buku Palasari- 82, Bandung: Informatika.

Nelson, Mark and Gailly, Jean-Loup. 1996. *The Data Compression Book (Second Edition)*, M&T Books

Salomon, D. (2007). *Data Compression*. Speinger.

Suriawati. (2019). China Diduga Menginstal Aplikasi Pengintai di Ponsel Pengunjung. Retrieved from Rakyatku News website: http://news.rakyatku.com/read/156402/2019/07/03/china-diduga-menginstal-aplikasi-pengintai-di-ponsel-pengunjung

Sutardi. (2014). *Implementasi dan analisis kinerja algoritma shannon- fano untuk kompresi file text*. *6*(1), 53–60.

Wibowo, I., Susanto, B., & Shamir, A. (2009). *Penerapan algoritma kriptografi asimetris rsa untuk keamanan data di oracle*. (1).

Wiryadinata, R. (2007). Data Compression Coding Using Static and Dynamic Method of Shannon-Fano Algorithm. *Jurnal Media Informatika, Vol. 5, No. 2*, *5*(2), 129–139. Retrieved from http://jurnal.uii.ac.id/index.php/media-informatika/article/viewFile/115/77

Wulansari, D., Setyawan, F. A., & Susanto, H. (2016). *Mengukur Kecepatan Enkripsi dan Dekripsi Algoritma RSA pada Pengembangan Sistem Informasi Text Security*. (Snik), 85–91.

Zulham, M., Kurniawan, H., & Rahmad, I. F. (2014). *Perancangan Aplikasi Keamanan Data Email Menggunakan Algoritma Enkripsi Rc6*. 96–101.

# DAFTAR RIWAYAT HIDUP

## DATA PRIBADI

| | |
|---|---|
| Nama | Jeprianto |
| Tempat Tgl Lahir | Bekasi, 28 Desember 1994 |
| Jenis Kelamin | Laki – Laki |
| Agama | Islam |
| Tinggi Badan | 176 cm |
| Berat Badan | 60 kg |
| Kewarganegaraan | Indonesia |

## LATAR BELAKANG PENDIDIKAN

| | |
|---|---|
| Tahun 2000-2003 | SDN 20 Jatiwarna |
| Tahun 2003-2004 | SDN 4 Ampalu Tinggi |
| Tahun 2004-2006 | SDN 046 Tanah Putih |
| Tahun 2006-2009 | MTSN Al-Kautsar Tanah Putih |
| Tahun 2009-2012 | SMAN 2 Tanah Putih |
| Tahun 2013-2019 | Teknik Informatika Universitas Islam Negeri Sultan Syarif Kasim Riau |

## INFORMASI TAMBAHAN

| | |
|---|---|
| E-Mail | jeprianto0@gmail.com |
| WhatsApp | 082283947483 |
| Line | manyihot |
| Instagram | jepriantoo |